

The pL^AT_ΕX 2_ε Sources

Ken Nakano & Japanese T_EX Development Community

2023-02-14 Patch level 1
(last updated: 2023/02/24)

Contents

a	plvers.dtx	1
1	pL^AT_ΕX 2_ε のバージョンの設定	1
2	起動時に実行するコード	2
2.1	L ^A T _Ε X 2 _ε 起動時の実行コードの取得	2
2.2	pL ^A T _Ε X 2 _ε 起動時に実行するコードの構築	2
2.2.1	L ^A T _Ε X 2 _ε 2023-06-01 以降の処理	2
2.2.2	L ^A T _Ε X 2 _ε 2022-11-15 以前の処理	3
2.3	フックシステムが利用可能かどうか	3
3	latexrelease パッケージへの対応	3
b	plexpl3.dtx	6
4	コード	6
5	pT_EX 系列の条件文	7
c	plfonts.dtx	8
6	概要	8
6.1	DOCSTRIP プログラムのためのオプション	8
6.2	拡張コマンド	8

7	コード	9
7.1	準備	10
7.1.1	和文フォント属性	10
7.1.2	長さ変数	11
7.1.3	一時コマンド	11
7.1.4	フォントリスト	12
7.1.5	支柱	13
7.2	NFSS2 の拡張コマンド	16
7.2.1	エンコードの宣言	16
7.2.2	ファミリの宣言	20
7.2.3	数式用フォント	28
7.2.4	従属書体の宣言	30
7.2.5	フォントの選択	32
7.2.6	エンコードの指定	40
7.2.7	ファミリの指定	42
7.2.8	シリーズの指定 (新 NFSS 対応)	44
7.2.9	シェイプの指定 (新 NFSS 対応)	48
7.2.10	書体の切り替え (新 NFSS 対応)	54
7.3	強調書体	68
7.4	下線マクロ	69
7.5	合成文字	69
7.6	イタリック補正と <code>\xkanjiskip</code>	76
7.7	デフォルト設定ファイルの読み込み	78
8	デフォルト設定ファイル	78
8.1	テキストフォント	79
8.2	プリロードフォント	80
8.3	組版パラメータ	80
9	フォント定義ファイル	81
d	plcore.dtx	83
10	概要	83
11	コード	83
11.1	プリアンブルコマンド	83
11.2	直前の JFM 由来スペースの削除【コミュニティ版独自】	84
11.3	改ページ	85
11.4	改行	86
11.5	オブジェクトの出力順序	88

11.6 トンボ	95
11.7 出力ルーチン	100
11.8 フロート環境	108
11.9 脚注マクロ	110
11.10 相互参照	114
11.11 疑似タイプ入力	115
11.12 tabbing 環境	117
11.13 用語集の出力	118
11.14 時分を示すカウンタ	118
11.15 tabular 環境	118
12 2013 年以降の新しい pT_EX 対応	122
13 e-pT_EX での FAM256 パッチの利用	124
14 L^AT_EX 2_ε と pL^AT_EX 2_ε の更新タイミングずれ対策	126
e plect.dtx	128
15 概要	128
16 組方向オプションについて	128
17 コード	129
17.1 表組環境	129
17.2 フロートとキャプションの出力位置	133
17.3 段落ボックス環境	138
17.4 作図環境	144
17.5 連数字／漢数字／傍点／下線	145
17.6 参照番号	147
f pl209.dtx	149
18 DOCSTRIP 用モジュール	149
19 2.09 互換マクロ	149
20 スタイルファイル	151
g kinsoku.dtx	153

21 禁則	153
21.1 半角文字に対する禁則	153
21.2 全角文字に対する禁則	153
22 文字間のスペース	155
22.1 ある英字と前後の漢字の間の制御	155
22.2 ある漢字と前後の英字の間の制御	158
 h jclasses.dtx	 160
23 オプションスイッチ	160
24 オプションの宣言	161
24.1 用紙オプション	161
24.2 サイズオプション	162
24.3 横置きオプション	162
24.4 トンボオプション	163
24.5 面付けオプション	163
24.6 組方向オプション	163
24.7 両面、片面オプション	163
24.8 二段組オプション	163
24.9 表題ページオプション	164
24.10 右左起こしオプション	164
24.11 数式のオプション	164
24.12 参考文献のオプション	164
24.13 日本語ファミリー宣言の抑制、和欧文両対応の数式文字	164
24.14 ドラフトオプション	165
24.15 オプションの実行	165
 25 フォント	 166
26 レイアウト	169
26.1 用紙サイズの決定	169
26.2 段落の形	170
26.3 ページレイアウト	170
26.3.1 縦方向のスペース	170
26.3.2 本文領域	171
26.3.3 マージン	176
26.4 脚注	179
26.5 フロート	180
26.5.1 フロートパラメータ	180

26.5.2	フロートオブジェクトの上限値	182
27	改ページ (日本語 \TeX 開発コミュニティ版のみ)	182
28	ページスタイル	184
28.1	マークについて	184
28.2	plain ページスタイル	185
28.3	jpl@in ページスタイル	185
28.4	headnombre ページスタイル	185
28.5	footnombre ページスタイル	186
28.6	headings スタイル	186
28.7	bothstyle スタイル	187
28.8	myheading スタイル	188
29	文書コマンド	189
29.1	表題	189
29.2	概要	193
29.3	章見出し	194
29.3.1	マークコマンド	194
29.3.2	カウンタの定義	194
29.3.3	前付け、本文、後付け	196
29.3.4	ボックスの組み立て	197
29.3.5	part レベル	198
29.3.6	chapter レベル	200
29.3.7	下位レベルの見出し	201
29.3.8	付録	202
29.4	リスト環境	203
29.4.1	enumerate 環境	205
29.4.2	itemize 環境	206
29.4.3	description 環境	207
29.4.4	verse 環境	207
29.4.5	quotation 環境	208
29.4.6	quote 環境	208
29.5	フロート	208
29.5.1	figure 環境	208
29.5.2	table 環境	209
29.6	キャプション	210
29.7	コマンドパラメータの設定	210
29.7.1	array と tabular 環境	210
29.7.2	tabbing 環境	211
29.7.3	minipage 環境	211

29.7.4	framebox 環境	211
29.7.5	equation と eqnarray 環境	211
30	フォントコマンド	211
31	相互参照	213
31.1	目次	213
31.1.1	本文目次	215
31.1.2	図目次と表目次	217
31.2	参考文献	218
31.3	索引	218
31.4	脚注	219
32	今日の日付	219
33	初期設定	220
i	jltxdoc.dtx	222

File a

plvers.dtx

1 pL^AT_EX 2_ε のバージョンの設定

現在の pL^AT_EX 2_ε がベースとした L^AT_EX 2_ε のバージョンは、下記のとおりです。

```
1 <(*2ekernel)
2 %\def\fmtname{LaTeX2e}
3 %\edef\fmtversion
4 </2ekernel>
5 <latexrelease>\edef\latexreleaseversion
6 <platexrelease>\edef\pknown@latexreleaseversion
7 <(*2ekernel | latexrelease | platexrelease)
8   {2021-11-15}
9 </2ekernel | latexrelease | platexrelease>
```

また、現在の pL^AT_EX 2_ε は最低でも L^AT_EX 2_ε 2017-04-15 以降（バージョン番号すなわち日付が YYYY/MM/DD 形式から YYYY-MM-DD 形式に変更された版）を前提とします。なお、L^AT_EX 2_ε 2017/01/01 以降は e-TeX 必須になっています。

```
10 <(*plcore)
11 \ifx\fmtversion\undefined
12   \errhelp{Please reinstall LaTeX.}%
13   \errmessage{This cannot happen!^^JYour file 'latex.ltx'
14               might be broken}\@@end
15 \else
16   \ifnum\expandafter\@parse@version\fmtversion//00\@nil<20170415
17     \errhelp{Please update your TeX installation; if not available,
18             obtain it^^Jmanually from CTAN
19             (https://ctan.org/pkg/latex-base) or from^^JGitHub
20             (https://github.com/latex3/latex2e).}%
21     \errmessage{This version of pLaTeX2e requires LaTeX2e 2017-04-15
22                 or newer!^^JObtain a newer version of 'latex',
23                 otherwise pLaTeX2e setup will^^Jnever succeed}\@@end
24   \fi
25 \fi
26 </plcore>
```

\pfmtname pL^AT_EX 2_ε のフォーマットファイル名とバージョンを定義します。

```
\pfmtversion 27 <(*plcore)
\ppatch@level 28 \def\pfmtname{pLaTeX2e}
                29 \def\pfmtversion
30 </plcore>
31 <platexrelease>\edef\platexreleaseversion
32 <(*plcore | platexrelease)
33   {2023-02-14}
34 </plcore | platexrelease>
35 <(*plcore)
36 \def\ppatch@level{1}
37 </plcore>
```

コミュニティ版 p \LaTeX 2 ϵ ではパッチファイルを使用しないので、パッチファイルをロードするコードは削除しました。

2 起動時に実行するコード

2.1 \LaTeX 2 ϵ 起動時の実行コードの取得

このファイルの直前で \LaTeX 2 ϵ の latex.ltx が読み込まれているはずなので、その起動時の実行コード (`\everyjob` トークンの内容) を保存します。

\LaTeX 2 ϵ 2018-04-01 patch level 1 までは、`\everyjob` が

```
\typeout{LaTeX2e version}\typeout{Babel version}
```

だけでしたが、patch level 2 以降ではいくつかのコードが `\everyjob` で遅延実行されるようになっています。それらのコードを抽出するため、最初と最後に区切りトークン (それぞれ `\platexNILa` と `\platexNILb`) を付けておきます。

なお、 \LaTeX 2 ϵ 2023-06-01 以降ではこのコードは使用されません。

```
38 (*plcore)
39 \edef\platexBANNER{\noexpand\platexNILa
40                    \the\everyjob % LaTeX banner and delayed codes
41                    \noexpand\platexNILb}
```

2.2 p \LaTeX 2 ϵ 起動時に実行するコードの構築

`\everyjob` \LaTeX 2 ϵ 起動時の実行コードを元に、p \LaTeX 2 ϵ 用の調整を加えます。

```
42 \begingroup
    p $\text{\LaTeX}$  2 $\epsilon$  のバージョン表示を作ります。
43 \ifnum\ppatch@level=0
44   \toks2={\pfmtname\space<\pfmtversion>}%
45 \else\ifnum\ppatch@level>0
46   \toks2={\pfmtname\space<\pfmtversion>+\ppatch@level}%
47 \else
48   \toks2={\pfmtname\space<\pfmtversion>-pre\ppatch@level}%
49 \fi\fi
```

2.2.1 \LaTeX 2 ϵ 2023-06-01 以降の処理

\LaTeX 2 ϵ 2023-06-01 以降では、 \LaTeX 2 ϵ のバージョン情報はトークンリスト `\LaTeXReleaseInfo` に格納されており、p \LaTeX 2 ϵ のバージョン情報をまとめて表示するためにはこのトークンリストを調整するだけで良くなりました。

以前と同じ

```
p $\text{\LaTeX}$ 2e <xxxx-yy-zz> (based on LaTeX2e ...)
```

という形式のでも良かったのですが、それでは `\LaTeXreleaseInfo` の中身をパースする必要が出てくるため

```
p $\text{\LaTeX}$ 2e <xxxx-yy-zz>, based on
LaTeX2e ...
```


と、単純に前置する形にしています。

なお、このコードでは `\expanded` プリミティブを使っています。

```
50 \ifdefined\LaTeXReleaseInfo
51   \global\LaTeXReleaseInfo\expandafter{\expanded{%
52     \noexpand\show@release@info{\the\toks2, based on}%
53     \unexpanded\expandafter{\the\LaTeXReleaseInfo}%
54   }}%
```

2.2.2 L^AT_EX 2_ε 2022-11-15 以前の処理

`\everyjob` の内容をパースして

- L^AT_EX 2_ε のバージョン表示の中身 (`\typeout{}` の引数) を #2
- バージョン表示の前に実行されるコードがあれば #1
- バージョン表示の後に残っているコードがあれば #3

に入れます。2020 年時点では #1 は空、#3 は欧文 inputenc の UTF-8 化で遅延されたコードに該当します。そして、L^AT_EX 2_ε のバージョンと pL^AT_EX 2_ε のバージョンをまとめて表示するように整形します。

```
55 \else
56   \edef\platexNILa#1\typeout#2#3\platexNILb{%
57     #1\noexpand\typeout{\the\toks2 \space(based on #2)}#3}
58   \global\everyjob\expandafter\expandafter\expandafter{\platexBANNER}%
```

不要になったマクロ定義は削除しておきます。

```
59 \fi
60 \endgroup
61 \let\platexBANNER=\@undefined
62 \</plcore>
```

2.3 フックシステムが利用可能かどうか

`\pltx@newhook@avail` フォーマット作成時 (`latex.ltx` の読込後すぐ) と、`platexrelease` パッケージ内 (`latexrelease` パッケージ読込後すぐ) でそれぞれ判定する必要があります。

```
63 < *plcore | plhookrelease >
64 \chardef\pltx@newhook@avail=\z@
65 \@ifl@t@r\fmtversion{2020/10/01}{\chardef\pltx@newhook@avail=\@ne}{ }
66 < /plcore | plhookrelease >
```

3 latexrelease パッケージへの対応

最後に、`latexrelease` パッケージへの対応です。

`\plIncludeInRelease` `platexrelease` パッケージでは `\plIncludeInRelease... \plEndIncludeInRelease` のブロックを使います。

```
67 < *plcore | platexrelease >
```

```

68 \newif\if@plincludeinrelease
69 \@plincludeinreleasefalse
70 \def\plIncludeInRelease#1{%
71   \if@plincludeinrelease
72     \PackageError{latexrelease}
73       {mis-matched \string\plIncludeInRelease}%
74       {There is an \string\plEndIncludeRelease\space missing}%
75     \@plincludeinreleasefalse
76   \fi
77   \kernel@ifnextchar[%
78     {\@plIncludeInRelease{#1}}
79     {\@plIncludeInRelease{#1}[#1]}]
80 \def\@plIncludeInRelease#1[#2]{\@plIncludeInRelease{#2}}
81 \def\@plIncludeInRelease#1#2#3{%
82   \toks@{[#1] #3}%
83   \expandafter\ifx\csname\string#2+\@currname+plIIR\endcsname\relax
84     \ifnum\expandafter\@parse@version#1//00\@nil
85       >\expandafter\@parse@version\pfmtversion//00\@nil
86       \GenericInfo{}{Skipping: \the\toks@}%
87       \expandafter\expandafter\expandafter\@gobble@plIncludeInRelease
88     \else
89       \GenericInfo{}{Applying: \the\toks@}%
90       \@plincludeinreleasetrue
91       \expandafter\let\csname\string#2+\@currname+plIIR\endcsname\@empty
92     \fi
93   \else
94     \GenericInfo{}{Already applied: \the\toks@}%
95     \expandafter\@gobble@plIncludeInRelease
96   \fi
97 }
98 \def\plEndIncludeInRelease{%
99   \if@plincludeinrelease
100     \@plincludeinreleasefalse
101   \else
102     \PackageError{latexrelease}
103       {mis-matched \string\plEndIncludeInRelease}{}%
104   \fi}
105 \long\def\@gobble@plIncludeInRelease#1\plEndIncludeInRelease{%
106   \@plincludeinreleasefalse
107   \@check@plIncludeInRelease#1\plIncludeInRelease
108   \@check@plIncludeInRelease\@end@check@plIncludeInRelease}
109 \long\def\@check@plIncludeInRelease#1\plIncludeInRelease
110   #2#3\@end@check@plIncludeInRelease{%
111   \ifx\@check@plIncludeInRelease#2\else
112     \PackageError{latexrelease}
113       {skipped \string\plIncludeInRelease\space for tag \string#2}{}%
114   \fi}
115 \plcore| latexrelease)

```

$\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$ が提供する latexrelease パッケージが読み込まれていて、かつ $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$ が提供する latexrelease パッケージが読み込まれていない場合は、巻き戻し機能に

よって pLaTeX 2_ε のコマンドが LaTeX 2_ε のコマンドで上書きされ、動作が壊れてしまいますので、警告を出します。

当初は \AtBeginDocument を使って \@begindocumenthook の末尾に警告文を入れていましたが、LaTeX 2_ε 2020-02-02 以降に付属の latexrelease パッケージで巻き戻すとフックの実行より早い段階（具体的には \process@table 内の \kanjiprocess@table 実行中）で「\series@maybe@drop@one@m が未定義」というエラーが出てしまうので、\process@table の先頭に警告文を入れます。万が一 \process@table も巻き戻し対象とされてしまった場合のため、\@begindocumenthook の先頭にも入れておきます。

LaTeX 2_ε 2020-10-01 以降では \process@table より早く実行されるフックが用意されたので、これを利用します。

```
116 (*plfinal)
117 \ifnum\pltx@newhook@avail=\z@
118 % for LaTeX2e 2020-02-02 PL5 or older
119 \expandafter\def\expandafter\process@table\expandafter{%
120   \expandafter\p@warn@latexrelease\process@table}
121 \begingroup
122 \toks@{\expandafter{\expandafter\p@warn@latexrelease\@begindocumenthook}}
123 \xdef\@begindocumenthook{\the\toks@}
124 \endgroup
125 \else
126 % for LaTeX2e 2020-10-01 or later
127 \AddToHook{begindocument/before}{\p@warn@latexrelease}
128 \fi
129 %
130 \def\p@warn@latexrelease{%
131   \ifx\latexreleaseversion\undefined\else
132     \ifx\platexreleaseversion\undefined
133       \@latex@warning@no@line{%
134         Package latexrelease is loaded.\MessageBreak
135         Some patches in pLaTeX2e core may be overwritten.\MessageBreak
136         Consider using platexrelease.\MessageBreak
137         See platex.pdf for detail}%
138     \fi
139   \fi
140   \let\p@warn@latexrelease\relax
141 }
142 </plfinal>
```

File b

plexpl3.dtx

L^AT_EX3 (expl3) で用意されていない「p_TE_X 系列の独自機能」を expl3 の文法で使えるようにするコードです。pL^AT_EX 2_ε 2020-10-01 で新設しました。

4 コード

パッケージとして宣言します。これで、pL^AT_EX 2_ε 2020-04-12 以前でも plexpl3.sty と plexpl3.ltx だけ入手すれば同等の機能が使えます。

```
1 (*package)
2 \NeedsTeXFormat{pLaTeX2e}
3 \RequirePackage{expl3}
4 \ProvidesExplPackage{plexpl3}{2020-09-28}{1.0}
5 {pTeX/upTeX-specific additions to expl3}
6 \endpackage
```

L^AT_EX 2_ε 2020-02-02 以降では expl3 が標準でフォーマットに読み込まれています。この場合は plexpl3 の機能をフォーマットに取り込みます。

```
7 \plcore\ifdefined\ExplSyntaxOn %--- expl3 available BEGIN
8 \plcore\ExplSyntaxOn
9 \plcore\package
10 \input plexpl3.ltx
11 \plcore\package
12 \plcore\ExplSyntaxOff
13 \plcore\fi %--- expl3 available END
```

platexrelease の roll-forward にも登録します。

```
14 \platexrelease\plIncludeInRelease{2020/10/01}%
15 \platexrelease {plexpl3}{Pre-load plexpl3}%
16 \platexrelease\RequirePackage{plexpl3}
17 \platexrelease\plEndIncludeInRelease
18 \platexrelease\plIncludeInRelease{0000/00/00}%
19 \platexrelease {plexpl3}{Not loading plexpl3}%
20 \platexrelease% Nothing to do
21 \platexrelease\plEndIncludeInRelease
```

以下のコードは plexpl3.ltx に書き出します。フォーマットとパッケージからの重複読み込みは禁止します。

```
22 (*code)
23 \cs_if_exist:NT \__platex_expl_loaded:
24 {
25   \GenericInfo{}
26   {Skipping:~ plexpl3~ code~ already~ part~ of~ the~ format}%
27   \endinput
28 }
29 \cs_new:Npn \__platex_expl_loaded: { }
```

5 pTeX 系列の条件文

pTeX 系列の条件文を expl3 の文法にします。

```
30 %% additions to l3box.dtx: writing directions (pTeX/upTeX-specific)
31 \cs_set_eq:NN \platex_direction_yoko: \tex_yoko:D
32 \cs_set_eq:NN \platex_direction_tate: \tex_tate:D
33 \cs_set_eq:NN \platex_direction_dtou: \tex_dtou:D
34 %
35 \prg_new_conditional:Npnn \platex_if_direction_yoko: { p, T, F, TF }
36   { \tex_ifydir:D \prg_return_true: \else: \prg_return_false: \fi: }
37 \prg_new_conditional:Npnn \platex_if_direction_tate: { p, T, F, TF }
38   { \tex_iftmdir:D \prg_return_true: \else: \prg_return_false: \fi: }
39 \prg_new_conditional:Npnn \platex_if_direction_dtou: { p, T, F, TF }
40   { \tex_ifddir:D \prg_return_true: \else: \prg_return_false: \fi: }
41 %
42 \prg_new_conditional:Npnn \platex_if_box_yoko:N #1 { p, T, F, TF }
43   { \tex_ifybox:D #1 \prg_return_true: \else: \prg_return_false: \fi: }
44 \prg_new_conditional:Npnn \platex_if_box_tate:N #1 { p, T, F, TF }
45   { \tex_iftbox:D #1 \prg_return_true: \else: \prg_return_false: \fi: }
46 \prg_new_conditional:Npnn \platex_if_box_dtou:N #1 { p, T, F, TF }
47   { \tex_ifdbox:D #1 \prg_return_true: \else: \prg_return_false: \fi: }
```

以上です。

```
48 \code
```

File c

plfonts.dtx

6 概要

ここでは、和文書体を NFSS2 のインターフェイスで選択するためのコマンドやマクロについて説明をしています。また、フォント定義ファイルや初期設定ファイルなどの説明もしています。新しいフォント選択コマンドの使い方については、`fntguide.tex` や `usrguide.tex` を参照してください。

第 6 節 この節です。このファイルの概要と DOCSTRIP プログラムのためのオプションを示しています。

第 7 節 実際のコードの部分です。

第 8 節 プリロードフォントやエラーフォントなどの初期設定について説明をしています。

第 9 節 フォント定義ファイルについて説明をしています。

6.1 DOCSTRIP プログラムのためのオプション

DOCSTRIP プログラムのためのオプションを次に示します。

オプション	意味
plcore	plcore.ltx の断片を生成します。
trace	ptrace.sty を生成します。
JY1mc	横組用、明朝体のフォント定義ファイルを生成します。
JY1gt	横組用、ゴシック体のフォント定義ファイルを生成します。
JT1mc	縦組用、明朝体のフォント定義ファイルを生成します。
JT1gt	縦組用、ゴシック体のフォント定義ファイルを生成します。
pldefs	pldefs.ltx を生成します。次の 4 つのオプションを付加することで、プリロードするフォントを選択することができます。デフォルトは 10pt です。
xpt	10pt プリロード
xipt	11pt プリロード
xiipt	12pt プリロード
ori	plfonts.tex に似たプリロード

6.2 拡張コマンド

pL^AT_EX 2_ε は、以下の新しいコマンドを定義します。

コマンド	意味
<code>\Declare{Yoko Tate}KanjiEncoding</code>	和文エンコードの宣言
<code>\DeclareKanjiEncodingDefaults</code>	デフォルトの和文エンコードの宣言
<code>\KanjiEncodingPair</code>	和文エンコードのセット化
<code>\DeclareKanjiFamily</code>	ファミリの宣言
<code>\DeclareKanjiSubstitution</code>	和文の代用フォントの宣言
<code>\DeclareErrorKanjiFont</code>	和文のエラーフォントの宣言
<code>\reDeclareMathAlphabet</code>	和欧文を同時に切り替えるコマンド宣言
<code>\{Declare Set}RelationFont</code>	従属書体の宣言
<code>\userelfont</code>	欧文書体を従属書体にする
<code>\adjustbaseline</code>	ベースラインシフト量の設定
<code>\{roman kanji}encoding</code>	エンコードの指定
<code>\{roman kanji}family</code>	ファミリの指定
<code>\{roman kanji}series[force]</code>	シリーズの指定
<code>\{roman kanji}shape[force]</code>	シェイプの指定
<code>\use{roman kanji}</code>	書体の切り替え
<code>\mcfamily, \gtfamily</code>	和文書体を明朝体、ゴシック体にする

さらに、オリジナルの \LaTeX 2_ϵ の以下のコマンドを再定義します。

コマンド	意味
<code>\DeclareFontEncoding</code>	エンコードの宣言
<code>\DeclareFontFamily</code>	ファミリの宣言
<code>\DeclareFixedFont</code>	フォントの名前の宣言
<code>\selectfont</code>	フォントを切り替える
<code>\set@fontsize</code>	フォントサイズの変更
<code>\fontencoding</code>	エンコードの指定
<code>\fontfamily</code>	ファミリの指定
<code>\fontseries[force]</code>	シリーズの指定
<code>\fontshape[force]</code>	シェイプの指定
<code>\usefont</code>	書体の切り替え
<code>\normalfont</code>	デフォルト値の設定に切り替える
<code>\bfseries, \mdseries</code>	シリーズを太字、中字にする

7 コード

この節で、実際のコードを説明します。

7.1 準備

NFSS2 を拡張するための準備です。和文フォントの属性を格納するオブジェクトや長さ変数、属性を切替える際の判断材料として使うリストなどを定義しています。

L^AT_EX の `tracefmt` パッケージに相当するデバッグ機能は、pL^AT_EX では `ptrace` パッケージで提供します。以前（アスキー版）では `ptrace` の前に `tracefmt` を手動で `\usepackage` する必要がありましたが、コミュニティ版では `ptrace` が自動で `tracefmt` を読み込むように改良してあります。

```
1 (*trace)
2 \NeedsTeXFormat{pLaTeX2e}
3 \ProvidesPackage{ptrace}
4   [2021/06/27 v1.7n Standard pLaTeX package (font tracing)]
5 \RequirePackageWithOptions{tracefmt}
6 \</trace>
```

7.1.1 和文フォント属性

ここでは、和文フォントの属性を格納するためのオブジェクトについて説明をしています。

`\k@encoding` 和文エンコードを示すオブジェクトです。`\ck@encoding` は、最後に選択された和文エンコード名を示しています。`\cy@encoding` と `\ct@encoding` はそれぞれ、最後に選択された、横組用と縦組用の和文エンコード名を示しています。
`\ct@encoding` ここでは単に「空」に初期化するだけにしています。

```
7 (*plcore)
8 \let\k@encoding\@empty
9 \let\ck@encoding\@empty
10 \let\cy@encoding\@empty
11 \let\ct@encoding\@empty
```

`\k@family` 和文書体のファミリーを示すオブジェクトです。

```
12 \let\k@family\@empty
```

`\k@series` 和文書体のシリーズを示すオブジェクトです。

```
13 \let\k@series\@empty
```

`\k@shape` 和文書体のシェイプを示すオブジェクトです。

```
14 \let\k@shape\@empty
```

`\curr@kfontshape` 現在の和文フォント名を示すオブジェクトです。

```
15 \def\curr@kfontshape{\k@encoding/\k@family/\k@series/\k@shape}
```

`\rel@fontshape` 関連付けされたフォント名を示すオブジェクトです。

```
16 \def\rel@fontshape{\f@encoding/\f@family/\f@series/\f@shape}
```


7.1.2 長さ変数

ここでは、和文フォントの幅や高さなどを格納する変数について説明をしています。

頭文字が大文字の変数は、ノーマルサイズの書体の大きさを、基準値となります。これらは、`jart10.clo` などの補助クラスファイルで設定されます。

小文字だけからなる変数は、フォントが変更されたときに (`\selectfont` 内で) 更新されます。

`\Cht` `\Cht` は基準となる和文フォントの文字の高さを示します。`\cht` は現在の和文フォントの文字の高さを示します。なお、この“高さ”はベースラインより上の長さです。

```
17 \newdimen\Cht
18 \newdimen\cht
```

`\Cdp` `\Cdp` は基準となる和文フォントの文字の深さを示します。`\cdp` は現在の和文フォントの文字の深さを示します。なお、この“深さ”はベースラインより下の長さです。

```
19 \newdimen\Cdp
20 \newdimen\cdp
```

`\Cwd` `\Cwd` は基準となる和文フォントの文字の幅を示します。`\cwd` は現在の和文フォントの文字の幅を示します。

```
21 \newdimen\Cwd
22 \newdimen\cwd
```

`\Cvs` `\Cvs` は基準となる行送りを示します。ノーマルサイズの `\baselineskip` と同値です。`\cvs` は現在の行送りを示します。

```
23 \newdimen\Cvs
24 \newdimen\cvs
```

`\Chs` `\Chs` は基準となる字送りを示します。`\Cwd` と同値です。`\chs` は現在の字送りを示します。

```
25 \newdimen\Chs
26 \newdimen\chs
```

`\cHT` `\cHT` は、現在のフォントの高さに深さを加えた長さを示します。`\set@fontsize` コマンド (実際は `\size@update`) で更新されます。

```
27 \newdimen\cHT
```

7.1.3 一時コマンド

`\afont` L^AT_EX 内部の `\do@subst@correction` マクロでは、`\fontname\font` で返される外部フォント名を用いて、L^AT_EX フォント名を定義しています。したがって、`\font` をそのまま使うと、和文フォント名に欧文の外部フォントが登録されたり、縦組フォント名に横組用の外部フォントが割り付けられたりしますので、`\jfont` か `\tfont` を用いるようにします。`\afont` は、`\font` コマンドの保存用です。

```
28 \let\afont\font
```

7.1.4 フォントリスト

ここでは、フォントのエンコードやファミリの名前を登録するリストについて説明をしています。

pL^AT_EX 2_ε の NFSS2 では、一つのコマンドで和文か欧文のいずれか、あるいは両方を変更するため、コマンドに指定された引数が何を示すのかを判断しなくてはなりません。この判断材料として、リストを用います。

このときの具体的な判断手順については、エンコード選択コマンドやファミリ選択コマンドなどの定義を参照してください。

`\inlist@` 次のコマンドは、エンコードやファミリのリスト内に第二引数で指定された文字列があるかどうかを調べるマクロです。結果は `\ifin@` に格納されます。第二引数はリストそのもの（リストが格納されたマクロではなく）を指定することになります。典型的には以下のように呼び出します。

```
\edef\tmp@item{\k@encoding}%
\expandafter\expandafter\expandafter
\inlist@\expandafter\tmp@item\expandafter{\kyenc@list}
```

`\do@subst@correction` の日本語化に必要なので、pL^AT_EX 2_ε 2020-04-12 以降では比較時に引数・リストとも `\detokenize` によって文字列化するようにしました。

```
29 </plcore>
30 <platexrelease>\plIncludeInRelease{2020/04/12}{\inlist@}
31 <platexrelease>          {Detokenize}%
32 <*plcore | platexrelease>
33 \def\inlist@#1#2{%
34   \edef\reserved@a{%
35     \unexpanded{\def\in@##1<}%
36     \detokenize{#1}%
37     \unexpanded{>##2##3\in@{\ifx\in@##2\in@false\else\in@true\fi}\in@}%
38     \detokenize{#2}%
39     \unexpanded{<}%
40     \detokenize{#1}%
41     \unexpanded{>\in@\in@}}%
42   \reserved@a}
43 </plcore | platexrelease>
44 <platexrelease>\plEndIncludeInRelease
45 <platexrelease>\plIncludeInRelease{0000/00/00}{\inlist@}
46 <platexrelease>          {ASCII Corporation original}%
47 <platexrelease>\def\inlist@#1#2{%
48   <platexrelease>   \def\in@##1<#1>##2##3\in@{%
49   <platexrelease>   \ifx\in@##2\in@false\else\in@true\fi}%
50 <platexrelease>   \in@##2<#1>\in@\in@}
51 <platexrelease>\plEndIncludeInRelease
52 <*plcore>
```

`\enc@elt` `\enc@elt` と `\fam@elt` は、登録されているエンコードに対して、なんらかの処理を `\fam@elt` 逐次的に行ないたいときに使用することができます。

```
53 \def\fam@elt{\noexpand\fam@elt}
54 \def\enc@elt{\noexpand\enc@elt}
```

`\fenc@list` `\fenc@list` には、`\DeclareFontEncoding` コマンドで宣言されたエンコード名が
`\kenc@list` 格納されていきます。

`\kyenc@list` `\kyenc@list` には、`\DeclareYokoKanjiEncoding` コマンドで宣言されたエン
`\ktenc@list` コード名が格納されていきます。`\ktenc@list` には、`\DeclareTateKanjiEncoding`
コマンドで宣言されたエンコード名が格納されていきます。

ここで、これらのリストに具体的な値を入れて初期化をするのは、リストにエン
コードの登録をするように `\DeclareFontEncoding` を再定義する前に、欧文エン
コードが宣言されるため、リストに登録されないからです。

```
55 \def\fenc@list{\enc@elt<OML>\enc@elt<T1>\enc@elt<OT1>\enc@elt<OMS>%  
56 \enc@elt<OMX>\enc@elt<TS1>\enc@elt<U>}  
57 \let\kenc@list\@empty  
58 \let\kyenc@list\@empty  
59 \let\ktenc@list\@empty
```

`\kfam@list` `\kfam@list` には、`\DeclareKanjiFamily` コマンドで宣言されたファミリ名が格納
`\ffam@list` されていきます。

`\notkfam@list` `\ffam@list` には、`\DeclareFontFamily` コマンドで宣言されたファミリ名が格
`\notffam@list` 納されていきます。

`\notkfam@list` には、和文ファミリではないと推測されたファミリ名が格納され
ていきます。このリストは `\fontfamily` コマンドで作成されます。

`\notffam@list` には欧文ファミリではないと推測されたファミリ名が格納されて
いきます。このリストは `\fontfamily` コマンドで作成されます。

ここで、これらのリストに具体的な値を入れて初期化をするのは、リストにファミ
リの登録をするように、`\DeclareFontFamily` が再定義される前に、このコマン
ドが使用されるため、リストに登録されないからです。

```
60 \def\kfam@list{\fam@elt<mc>\fam@elt<gt>}  
61 \def\ffam@list{\fam@elt<cmr>\fam@elt<cmss>\fam@elt<cmtt>%  
62 \fam@elt<cmn>\fam@elt<cmsy>\fam@elt<cmex>}
```

つぎの二つのリストの初期値として、上記の値を用います。これらのファミリ名は、
和文でないこと、欧文でないことがはっきりしています。

```
63 \let\notkfam@list\ffam@list  
64 \let\notffam@list\kfam@list
```

7.1.5 支柱

行間の調整などに用いる支柱です。支柱のもととなるボックスの大きさは、フォン
トサイズが変更されるたびに、`\set@fontsize` コマンドによって更新されます。

コミュニティ版 p_{La}T_EX 2_ε 2017/04/08 での変更：従来、横組ボックス用の支柱は
`\strutbox` で、高さや深さが 7 対 3 となっていました。これは p_{La}T_EX 単体では問
題になりませんでしたが、海外製の L_AT_EX パッケージを縦組で使った場合に、意
図しない幅や高さ取得されることがありました。この不都合を回避するため、コ
ミュニティ版 p_{La}T_EX では次の方法をとります。

- `\ystrutbox` (新設)：高さや深さが 7 対 3 の横組用の支柱ボックスレジスタ

- `\tstrutbox`：高さ と 深さが 5 対 5 の縦組用の支柱ボックスレジスタ
- `\zstrutbox`：高さ と 深さが 7 対 3 の縦組用の支柱ボックスレジスタ
- `\strutbox`（仕様変更）：縦横のディレクションに応じて `\tstrutbox` または `\ystrutbox` に展開されるマクロ

すなわち、従来の \LaTeX における `\strutbox` と同じ挙動を示すのが、新設された `\ystrutbox` ということになります。

`\tstrutbox` `\tstrutbox` は高さ と 深さが 5 対 5、`\zstrutbox` は高さ と 深さが 7 対 3 の支柱ボックスとなります。これらは縦組ボックスの行間の調整などに使います。

```
65 \newbox\tstrutbox
66 \newbox\zstrutbox
```

`\ystrutbox` `\ystrutbox` は高さ と 深さが 7 対 3 の横組用の支柱ボックスです。

```
67 </plcore>
68 <latexrelease>\plIncludeInRelease{2017/04/08}{\ystrutbox}
69 <latexrelease>          {Add \ystrutbox}%
70 <*plcore | latexrelease>
71 \newbox\ystrutbox
72 </plcore | latexrelease>
73 <latexrelease>\plEndIncludeInRelease
74 <latexrelease>\plIncludeInRelease{0000/00/00}{\ystrutbox}
75 <latexrelease>          {Add \ystrutbox}%
76 <latexrelease>\let\ystrutbox\undefined
77 <latexrelease>\plEndIncludeInRelease
```

`\strutbox` `\strutbox` は縦横両対応です。

```
78 <latexrelease>\plIncludeInRelease{2017/04/08}{\strutbox}
79 <latexrelease>          {Macro definition of \strutbox}%
80 <*plcore | latexrelease>
81 \def\strutbox{\iftdir\tstrutbox\else\ystrutbox\fi}
82 </plcore | latexrelease>
83 <latexrelease>\plEndIncludeInRelease
84 <latexrelease>\plIncludeInRelease{0000/00/00}{\strutbox}
85 <latexrelease>          {LaTeX2e original}%
86 <latexrelease>\newbox\strutbox % emulation purpose only
87 <latexrelease>\plEndIncludeInRelease
```

`\strut` ディレクションに応じて `\ystrutbox` と `\tstrutbox` を使い分けます。オリジナルの \LaTeX では `ltplain.dtx` で定義されていますが、 \LaTeX 2_ε 2019-10-01 以降ではさらに `ltdfns.dtx` で `\MakeRobust` を前置されるため、`robust` になります。

```
88 <latexrelease>\plIncludeInRelease{2019/10/01}{\strut}
89 <latexrelease>          {Make robust}%
90 <*plcore | latexrelease>
91 \DeclareRobustCommand\strut{\relax
92   \iftdir
93     \ifmmode\copy\tstrutbox\else\unhcopy\tstrutbox\fi
94   \else
```

```

95     \ifmmode\copy\ystrutbox\else\unhcopy\ystrutbox\fi
96   \fi}
97 \
```

plcore | latexrelease)

```

98 \plEndIncludeInRelease
99 \plIncludeInRelease{2017/04/08}{\strut}
100 \plIncludeInRelease{2017/04/08}{\strut}
101 \plIncludeInRelease{2017/04/08}{\strut}
102 \plIncludeInRelease{2017/04/08}{\strut}
103 \plIncludeInRelease{2017/04/08}{\strut}
104 \plIncludeInRelease{2017/04/08}{\strut}
105 \plIncludeInRelease{2017/04/08}{\strut}
106 \plIncludeInRelease{2017/04/08}{\strut}
107 \plIncludeInRelease{2017/04/08}{\strut}
108 \plIncludeInRelease{2017/04/08}{\strut}
109 \plIncludeInRelease{2017/04/08}{\strut}
110 \plIncludeInRelease{2017/04/08}{\strut}
111 \plIncludeInRelease{2017/04/08}{\strut}
112 \plIncludeInRelease{2017/04/08}{\strut}
113 \plIncludeInRelease{2017/04/08}{\strut}
114 \plIncludeInRelease{2017/04/08}{\strut}
115 \plIncludeInRelease{2017/04/08}{\strut}
116 \plIncludeInRelease{2017/04/08}{\strut}
117 \plIncludeInRelease{2017/04/08}{\strut}
118 \plIncludeInRelease{2017/04/08}{\strut}

\tsstrut
\zstrut
119 \plIncludeInRelease{2019/10/01}{\tsstrut}
120 \plIncludeInRelease{2019/10/01}{\tsstrut}
121 \plIncludeInRelease{2019/10/01}{\tsstrut}
122 \plIncludeInRelease{2019/10/01}{\tsstrut}
123 \plIncludeInRelease{2019/10/01}{\tsstrut}
124 \plIncludeInRelease{2019/10/01}{\tsstrut}
125 \plIncludeInRelease{2019/10/01}{\tsstrut}
126 \plIncludeInRelease{2019/10/01}{\tsstrut}
127 \plIncludeInRelease{2019/10/01}{\tsstrut}
128 \plIncludeInRelease{2019/10/01}{\tsstrut}
129 \plIncludeInRelease{2019/10/01}{\tsstrut}
130 \plIncludeInRelease{2019/10/01}{\tsstrut}
131 \plIncludeInRelease{2019/10/01}{\tsstrut}
132 \plIncludeInRelease{2019/10/01}{\tsstrut}
133 \plIncludeInRelease{2019/10/01}{\tsstrut}
134 \plIncludeInRelease{2019/10/01}{\tsstrut}
135 \plIncludeInRelease{2019/10/01}{\tsstrut}
136 \plIncludeInRelease{2019/10/01}{\tsstrut}

\ystrut
137 \plIncludeInRelease{2019/10/01}{\ystrut}
138 \plIncludeInRelease{2019/10/01}{\ystrut}
139 \plIncludeInRelease{2019/10/01}{\ystrut}
140 \plIncludeInRelease{2019/10/01}{\ystrut}
141 \plIncludeInRelease{2019/10/01}{\ystrut}
142 \plIncludeInRelease{2019/10/01}{\ystrut}
143 \plIncludeInRelease{2019/10/01}{\ystrut}

```

```

144 <platexrelease>\plIncludeInRelease{2017/04/08}{\ystrut}
145 <platexrelease>          {Add \ystrut}%
146 <platexrelease>\def\ystrut{\relax\hbox{\yoko
147 <platexrelease>      \ifmmode\copy\ystrutbox\else\unhcopy\ystrutbox\fi}}
148 <platexrelease>\expandafter \let \csname ystrut \endcsname \@undefined
149 <platexrelease>\plEndIncludeInRelease
150 <platexrelease>\plIncludeInRelease{0000/00/00}{\ystrut}
151 <platexrelease>          {Add \ystrut}%
152 <platexrelease>\let\ystrut\@undefined
153 <platexrelease>\expandafter \let \csname ystrut \endcsname \@undefined
154 <platexrelease>\plEndIncludeInRelease
155 <*plcore>

```

7.2 NFSS2 の拡張コマンド

NFSS2 の拡張コマンドを定義します。

7.2.1 エンコードの宣言

`\DeclareFontEncoding` 欧文エンコードを宣言するためのコマンドです。l^{at}ex_{2_ε}で定義されている
`\DeclareFontEncoding@` ものを、`\fenc@list` を作るように再定義をしています。

```

156 \def\DeclareFontEncoding{%
157   \begingroup
158   \nfss@catcodes
159   \expandafter\endgroup
160   \DeclareFontEncoding@}
161 </plcore>
162 <platexrelease>\plIncludeInRelease{2018/04/01}{\DeclareFontEncoding@}
163 <platexrelease>          {UTF-8 Encoding}%
164 <*plcore | platexrelease>

```

まず、L^AT_EX 2_ε 2017-04-15 以前の場合のコードです。このコードは、`\UseRawInputEncoding` の内部でも使われます。

```

165 % for compatibility with LaTeX2e 2017-04-15 or earlier.
166 % this code is used if MLTeX is enabled
167 \def\DeclareFontEncoding@#1#2#3{%
168   \expandafter
169   \ifx\csname T@#1\endcsname\relax
170     \def\cdp@elt{\noexpand\cdp@elt}%
171     \xdef\cdp@list{\cdp@list\cdp@elt{#1}%
172                   {\default@family}{\default@series}%
173                   {\default@shape}}%
174     \expandafter\let\csname#1-cmd\endcsname\@changed@cmd

```

以下の 2 行が pL^AT_EX 2_ε による追加部分です。

```

175   \def\enc@elt{\noexpand\enc@elt}%
176   \xdef\fenc@list{\fenc@list\enc@elt<#1>}%
177   \else
178     \@font@info{Redeclaring font encoding #1}%
179   \fi
180   \global\@namedef{T@#1}{#2}%
181   \global\@namedef{M@#1}{\default@M#3}%

```

```

182 \xdef\LastDeclaredEncoding{#1}%
183 }
184 \let\DeclareFontEncoding@saved\DeclareFontEncoding@

```

次に、 $\text{\LaTeX 2}_{\epsilon}$ 2018-04-01 以降の場合のコードです。

```

185 \ifx\IeC\@undefined\else
186 % for LaTeX2e with UTF-8 input.
187 \def\DeclareFontEncoding@#1#2#3{%
188   \expandafter
189   \ifx\csname T@#1\endcsname\relax
190     \def\cdp@elt{\noexpand\cdp@elt}%
191     \xdef\cdp@list{\cdp@list\cdp@elt{#1}%
192                  {\default@family}{\default@series}%
193                  {\default@shape}}%
194     \expandafter\let\csname#1-cmd\endcsname\@changed@cmd

```

$\text{\LaTeX 2}_{\epsilon}$ 2018-04-01 で、既定の欧文入力エンコーディングが UTF-8 になりました。これは、`latex.ltx` が `utf8.def` (従来は \LaTeX ソースに `\usepackage[utf8]{inputenc}` と書いたときに読み込まれていたもの) を読み込むことで実現されています。`utf8.def` は `\DeclareFontEncoding@` を再定義するので、これに合わせるためのコードを追加します。

```

195   \begingroup
196     \wlog{Now handling font encoding #1 ...}%
197     \lowercase{%
198       \InputIfFileExists{#1enc.dfu}}%
199       {\wlog{... processing UTF-8 mapping file for font %
200            encoding #1}}%
201       {\wlog{... no UTF-8 mapping file for font encoding #1}}%
202   \endgroup

```

以下の 2 行が $\text{\LaTeX 2}_{\epsilon}$ による追加部分です。

```

203   \def\enc@elt{\noexpand\enc@elt}%
204   \xdef\fenc@list{\fenc@list\enc@elt<#1>}%
205   \else
206     \@font@info{Redefining font encoding #1}%
207     \fi
208     \global\@namedef{T@#1}{#2}%
209     \global\@namedef{M@#1}{\default@M#3}%
210     \xdef\LastDeclaredEncoding{#1}%
211   }

212 \fi
213 </plcore | platexrelease>
214 <platexrelease>\plEndIncludeInRelease
215 <platexrelease>\plIncludeInRelease{0000/00/00}{\DeclareFontEncoding@}
216 <platexrelease>          {ASCII Corporation original}%
217 <platexrelease>\def\DeclareFontEncoding@#1#2#3{%
218 <platexrelease>  \expandafter
219 <platexrelease>  \ifx\csname T@#1\endcsname\relax
220 <platexrelease>    \def\cdp@elt{\noexpand\cdp@elt}%
221 <platexrelease>    \xdef\cdp@list{\cdp@list\cdp@elt{#1}%
222 <platexrelease>                  {\default@family}{\default@series}%
223 <platexrelease>                  {\default@shape}}%

```

```

224 <latexrelease> \expandafter\let\csname#1-cmd\endcsname\@changed@cmd
225 <latexrelease> \def\enc@elt{\noexpand\enc@elt}%
226 <latexrelease> \xdef\fenc@list{\fenc@list\enc@elt<#1>}%
227 <latexrelease> \else
228 <latexrelease> \@font@info{Redefining font encoding #1}%
229 <latexrelease> \fi
230 <latexrelease> \global\@namedef{T@#1}{#2}%
231 <latexrelease> \global\@namedef{M@#1}{\default@M#3}%
232 <latexrelease> \xdef\LastDeclaredEncoding{#1}%
233 <latexrelease> }
234 <latexrelease>\let\DeclareFontEncoding@saved\@undefined
235 <latexrelease>\plEndIncludeInRelease
236 <*plcore>

```

\DeclareKanjiEncoding 和文エンコードの宣言をするコマンドです。

```

\DeclareYokoKanjiEncoding 237 \def\DeclareKanjiEncoding#1{%
\DeclareYokoKanjiEncoding@ 238 \@latex@warning{%
\DeclareTateKanjiEncoding 239 The \string\DeclareKanjiEncoding\space is obsoleted command. Please use
\DeclareTateKanjiEncoding 240 \MessageBreak
\DeclareTateKanjiEncoding@ 241 the \string\DeclareTateKanjiEncoding\space for ‘Tate-kumi’ encoding, and
242 \MessageBreak
243 the \string\DeclareYokoKanjiEncoding\space for ‘Yoko-kumi’ encoding.
244 \MessageBreak
245 I treat the ‘#1’ encoding as ‘Yoko-kumi’..}
246 \DeclareYokoKanjiEncoding{#1}%
247 }
248 \def\DeclareYokoKanjiEncoding{%
249 \begingroup
250 \nfss@catcodes
251 \expandafter\endgroup
252 \DeclareYokoKanjiEncoding@}
253 %
254 \def\DeclareYokoKanjiEncoding@#1#2#3{%
255 \expandafter
256 \ifx\csname T@#1\endcsname\relax
257 \def\cdp@elt{\noexpand\cdp@elt}%
258 \xdef\cdp@list{\cdp@list\cdp@elt{#1}%
259 {\default@k@family}{\default@k@series}%
260 {\default@k@shape}}%
261 \expandafter\let\csname#1-cmd\endcsname\@changed@kcmd
262 \def\enc@elt{\noexpand\enc@elt}%
263 \xdef\kyenc@list{\kyenc@list\enc@elt<#1>}%
264 \xdef\kenc@list{\kenc@list\enc@elt<#1>}%
265 \else
266 \@font@info{Redefining KANJI (yoko) font encoding #1}%
267 \fi
268 \global\@namedef{T@#1}{#2}%
269 \global\@namedef{M@#1}{\default@KM#3}%
270 }
271 %
272 \def\DeclareTateKanjiEncoding{%
273 \begingroup
274 \nfss@catcodes
275 \expandafter\endgroup

```



```

276 \DeclareTateKanjiEncoding@
277 %
278 \def\DeclareTateKanjiEncoding@#1#2#3{%
279 \expandafter
280 \ifx\csname T@#1\endcsname\relax
281 \def\cdp@elt{\noexpand\cdp@elt}%
282 \xdef\cdp@list{\cdp@list\cdp@elt{#1}%
283 {\default@k@family}{\default@k@series}%
284 {\default@k@shape}}%
285 \expandafter\let\csname#1-cmd\endcsname\@changed@kcmd
286 \def\enc@elt{\noexpand\enc@elt}%
287 \xdef\ktenc@list{\ktenc@list\enc@elt<#1>}%
288 \xdef\kenc@list{\kenc@list\enc@elt<#1>}%
289 \else
290 \@font@info{Redeclaring KANJI (tate) font encoding #1}%
291 \fi
292 \global\@namedef{T@#1}{#2}%
293 \global\@namedef{M@#1}{\default@KM#3}%
294 }
295 %
296 \@onlypreamble\DeclareKanjiEncoding
297 \@onlypreamble\DeclareYokoKanjiEncoding
298 \@onlypreamble\DeclareYokoKanjiEncoding@
299 \@onlypreamble\DeclareTateKanjiEncoding
300 \@onlypreamble\DeclareTateKanjiEncoding@

```

`\DeclareKanjiEncodingDefaults` 和文エンコードのデフォルト値を宣言するコマンドです。`\DeclareFontEncodingDefaults` に相当します。

```

301 \def\DeclareKanjiEncodingDefaults#1#2{%
302 \ifx\relax#1\else
303 \ifx\default@KT\@empty\else
304 \@font@info{Overwriting KANJI encoding scheme text defaults}%
305 \fi
306 \gdef\default@KT{#1}%
307 \fi
308 \ifx\relax#2\else
309 \ifx\default@KM\@empty\else
310 \@font@info{Overwriting KANJI encoding scheme math defaults}%
311 \fi
312 \gdef\default@KM{#2}%
313 \fi}
314 \let\default@KT\@empty
315 \let\default@KM\@empty
316 \@onlypreamble\DeclareKanjiEncodingDefaults

```

`\KanjiEncodingPair` 和文の縦横のエンコーディングはそれぞれ対にして扱うため、セット化するためのコマンドを定義します。第一引数が横組用、第二引数が縦組用です。

```

317 \def\KanjiEncodingPair#1#2{\@namedef{t@enc@#1}{#2}\@namedef{y@enc@#2}{#1}}

```

`\ensure@KanjiEncodingPair` 横書きと縦書きのエンコーディングは必ず `\KanjiEncodingPair` でセット化しないと使えません。もしセット化されていなければ、明快なエラーで知らせます。

```

318 </plcore>

```

```

319 <platexrelease>\plIncludeInRelease{2020/04/12}{\ensure@KanjiEncodingPair}
320 <platexrelease>          {Check \KanjiEncodingPair}%
321 <*plcore| platexrelease>
322 \def\ensure@KanjiEncodingPair#1{%
323   \edef\reserved@a{\csname #1@enc@k@encoding\endcsname}%
324   \edef\reserved@b{\csname c#1@encoding\endcsname}%

\reserved@a は、セット化が有効ならエンコードを表す文字トークン列、無効なら
\relax と同義の制御綴に展開されるマクロです。ここで、\ifcat（展開不能ト
ークンが現れるまで展開してから比較）を使います。

325   \ifcat\relax\reserved@a
326     \@latex@error
327       {KANJI Encoding pair for ‘\k@encoding’ undefined}%
328       {Use \string\KanjiEncodingPair, falling back to ‘\reserved@b’...}%
329     \expandafter\edef\reserved@a{\reserved@b}%
330   \fi}
331 </plcore| platexrelease>
332 <platexrelease>\plEndIncludeInRelease
333 <platexrelease>\plIncludeInRelease{0000/00/00}{\ensure@KanjiEncodingPair}
334 <platexrelease>          {ASCII Corporation original}%
335 <platexrelease>\let\ensure@KanjiEncodingPair\@undefined
336 <platexrelease>\plEndIncludeInRelease
337 <*plcore>

```

7.2.2 ファミリの宣言

`\DeclareFontFamily` 欧文ファミリを宣言するためのコマンドです。`\ffam@list` を作るように再定義をします。

```

338 \def\DeclareFontFamily#1#2#3{%
339   \ifundefined{T@#1}%
340     {\@latex@error{Encoding scheme ‘#1’ unknown}\@eha}%
341     {\edef\tmp@item{{#2}}%
342       \expandafter\expandafter\expandafter
343       \inlist@\expandafter\tmp@item\expandafter{\ffam@list}%
344       \ifin@ \else
345         \def\fam@elt{\noexpand\fam@elt}%
346         \xdef\ffam@list{\ffam@list\fam@elt<#2>}%
347       \fi
348       \def\reserved@a{#3}%
349       \global
350       \expandafter\let\csname #1+#2\expandafter\endcsname
351         \ifx \reserved@a@empty
352           \@empty
353         \else \reserved@a
354       \fi
355     }%
356 }

```

`\DeclareKanjiFamily` 和文ファミリを宣言するためのコマンドです。

```

357 \def\DeclareKanjiFamily#1#2#3{%
358   \ifundefined{T@#1}%
359     {\@latex@error{KANJI Encoding scheme ‘#1’ unknown}\@eha}%

```

```

360 {\edef\tmp@item{#2}}%
361 \expandafter\expandafter\expandafter
362 \inlist@\expandafter\tmp@item\expandafter{\kfam@list}%
363 \ifin@ \else
364 \def\fam@elt{\noexpand\fam@elt}%
365 \xdef\kfam@list{\kfam@list\fam@elt<#2>}%
366 \fi
367 \def\reserved@a{#3}%
368 \global
369 \expandafter\let\csname #1+#2\expandafter\endcsname
370 \ifx \reserved@a\@empty
371 \@empty
372 \else \reserved@a
373 \fi
374 }%
375 }

```

\DeclareKanjiSubstitution 目的の和文フォントが見つからなかったときに使う代用書体の宣言をするコマンドです。 \DeclareFontSubstitution に相当します。

```

376 </plcore>
377 <latexrelease>\plIncludeInRelease{2020/04/12}{\DeclareKanjiSubstitution}
378 <latexrelease> {Use \default@k@family etc.}%
379 <*plcore | latexrelease>
380 \def\DeclareKanjiSubstitution#1#2#3#4{%
381 \expandafter\ifx\csname T@#1\endcsname\relax
382 \@latex@error{KANJI Encoding scheme ‘#1’ unknown}\@eha
383 \else
384 \begingroup
385 \def\reserved@a{#1}%
386 \toks@{}%
387 \def\cdp@elt##1##2##3##4{%
388 \def\reserved@b{##1}%
389 \ifx\reserved@a\reserved@b
390 \addto@hook\toks@{\cdp@elt{#1}{#2}{#3}{#4}}%
391 \else
392 \addto@hook\toks@{\cdp@elt{##1}{##2}{##3}{##4}}%
393 \fi}%
394 \cdp@list
395 \xdef\cdp@list{\the\toks@}%
396 \endgroup
397 \global\@namedef{D@#1}{\def\default@k@family{#2}% !!!
398 \def\default@k@series{#3}% !!!
399 \def\default@k@shape{#4}}% !!!
400 \fi}
401 </plcore | latexrelease>
402 <latexrelease>\plEndIncludeInRelease
403 <latexrelease>\plIncludeInRelease{0000/00/00}{\DeclareKanjiSubstitution}
404 <latexrelease> {ASCII Corporation original}%
405 <latexrelease>\def\DeclareKanjiSubstitution#1#2#3#4{%
406 <latexrelease> \expandafter\ifx\csname T@#1\endcsname\relax
407 <latexrelease> \@latex@error{KANJI Encoding scheme ‘#1’ unknown}\@eha
408 <latexrelease> \else
409 <latexrelease> \begingroup
410 <latexrelease> \def\reserved@a{#1}%

```

```

411 <platexrelease>      \toks@{%
412 <platexrelease>      \def\cdp@elt##1##2##3##4{%
413 <platexrelease>      \def\reserved@a{##1}%
414 <platexrelease>      \ifx\reserved@a\reserved@b
415 <platexrelease>      \addto@hook\toks@{\cdp@elt{##1}{##2}{##3}{##4}}%
416 <platexrelease>      \else
417 <platexrelease>      \addto@hook\toks@{\cdp@elt{##1}{##2}{##3}{##4}}%
418 <platexrelease>      \fi}%
419 <platexrelease>      \cdp@list
420 <platexrelease>      \xdef\cdp@list{\the\toks@}%
421 <platexrelease>      \endgroup
422 <platexrelease>      \global\@namedef{D@#1}{\def\default@family{##2}%
423 <platexrelease>      \def\default@series{##3}%
424 <platexrelease>      \def\default@shape{##4}}%
425 <platexrelease>      \fi}
426 <platexrelease>\plEndIncludeInRelease
427 <platexrelease>% !!! Special case BEGIN
428 <platexrelease>% required for any emulation date
429 <platexrelease>% copied from (u)pldefs.ltx
430 <platexrelease>\def\pltx@tempa{JY1}\ifx\pltx@tempa\k@encoding
431 <platexrelease>\DeclareKanjiSubstitution{JY1}{mc}{m}{n}
432 <platexrelease>\DeclareKanjiSubstitution{JT1}{mc}{m}{n}
433 <platexrelease>\else\def\pltx@tempa{JY2}\ifx\pltx@tempa\k@encoding
434 <platexrelease>\DeclareKanjiSubstitution{JY2}{mc}{m}{n}
435 <platexrelease>\DeclareKanjiSubstitution{JT2}{mc}{m}{n}
436 <platexrelease>\fi\fi
437 <platexrelease>% emulate execution of \enc@update in \selectfont
438 <platexrelease>% before (u)pldefs.ltx is loaded
439 <platexrelease>\csname D@f@encoding\endcsname
440 <platexrelease>% emulate execution of \kenc@update in \selectfont
441 <platexrelease>% inside (u)pldefs.ltx
442 <platexrelease>\csname D@k@encoding\endcsname
443 <platexrelease>% !!! Special case END
444 <*plcore>
445 \@onlypreamble\DeclareKanjiSubstitution

```

\DeclareErrorKanjiFont \DeclareErrorFont に対応するコマンドです。代用書体で示された書体も見つからなかったときに最後の手段として使われるエラー書体を定義します。

```

446 </plcore>
447 <platexrelease>\plIncludeInRelease{2019/10/01}{\DeclareErrorKanjiFont}
448 <platexrelease>      {No side effects please}%
449 <*plcore | platexrelease>
450 \def\DeclareErrorKanjiFont#1#2#3#4#5{%
451   \xdef\error@kfontshape{%
452     \noexpand\expandafter\noexpand\split@name\noexpand\string
453     \expandafter\noexpand\csname#1/#2/#3/#4/#5\endcsname
454     \noexpand\@nil}%
455   \gdef\default@k@family{##2}%
456   \gdef\default@k@series{##3}%
457   \gdef\default@k@shape{##4}%
458   }
459 </plcore | platexrelease>
460 <platexrelease>\plEndIncludeInRelease
461 <platexrelease>\plIncludeInRelease{0000/00/00}{\DeclareErrorKanjiFont}

```

```

462 <latexrelease> {ASCII Corporation original}%
463 <latexrelease> \def\DeclareErrorKanjiFont#1#2#3#4#5{%
464 <latexrelease> \xdef\error@kfontshape{%
465 <latexrelease> \noexpand\expandafter\noexpand\split@name\noexpand\string
466 <latexrelease> \expandafter\noexpand\csname#1/#2/#3/#4/#5\endcsname
467 <latexrelease> \noexpand\@nil}%
468 <latexrelease> \gdef\default@k@family{#2}%
469 <latexrelease> \gdef\default@k@series{#3}%
470 <latexrelease> \gdef\default@k@shape{#4}%
471 <latexrelease> \global\let\k@family\default@k@family
472 <latexrelease> \global\let\k@series\default@k@series
473 <latexrelease> \global\let\k@shape\default@k@shape
474 <latexrelease> \gdef\font@size{#5}%
475 <latexrelease> \gdef\font@baselineskip{#5pt}}
476 <latexrelease> \plEndIncludeInRelease
477 <*plcore>
478 \@onlypreamble\DeclareErrorKanjiFont

```

`\wrong@fontshape` `\wrong@fontshape` を和文対応にします。`\DeclareKanjiSubstitution` で `\default@k@...`
`\wrong@al@fontshape` を使用する改良と同時になければなりません。
`\wrong@ja@fontshape` オリジナルの L^AT_EX の定義は、欧文用として使います。

```

479 </plcore>
480 <latexrelease> \plIncludeInRelease{2020/04/12}{\wrong@fontshape}
481 <latexrelease> {Japanese \wrong@fontshape}%
482 <*plcore | latexrelease>
483 \def\wrong@al@fontshape{%
484 \csname D@f@encoding\endcsname % install defaults if in math
485 \edef\reserved@a{\csname\curr@fontshape\endcsname}%
486 \ifx\last@fontshape\reserved@a
487 \errmessage{Corrupted NFSS tables}%
488 \error@fontshape
489 \else
490 \let\font@shape\default@shape
491 \expandafter\ifx\csname\curr@fontshape\endcsname\relax
492 \let\font@series\default@series
493 \expandafter
494 \ifx\csname\curr@fontshape\endcsname\relax
495 \let\font@family\default@family
496 \begingroup
497 \try@load@fontshape
498 \endgroup
499 \fi \fi
500 \fi
501 \@font@warning{Font shape '\expandafter\string\reserved@a'
502 \expandafter\@gobble\string\@undefined\MessageBreak
503 using '\curr@fontshape' instead\@wrong@font@char}%
504 \global\let\last@fontshape\reserved@a
505 \gdef\@defaultsubs{%
506 \@font@warning{Some font shapes were not available, defaults
507 substituted.\@gobbletwo}}%
508 \global\expandafter\expandafter\expandafter\let
509 \expandafter\reserved@a
510 \csname\curr@fontshape\endcsname

```

```

511 \xdef\font@name{%
512 \csname\curr@fontshape/\f@size\endcsname}%
513 \pickup@font}

```

和文用の定義です。

```

514 \def\wrong@ja@fontshape{%
515 \csname D@f@encoding\endcsname % install defaults if in math
516 \edef\reserved@a{\csname\curr@fontshape\endcsname}%
517 \ifx\last@fontshape\reserved@a
518 \errmessage{Corrupted NFSS tables}%
519 \error@fontshape
520 \else
521 \let\f@shape\default@k@shape % !!!
522 \expandafter\ifx\csname\curr@fontshape\endcsname\relax
523 \let\f@series\default@k@series % !!!
524 \expandafter
525 \ifx\csname\curr@fontshape\endcsname\relax
526 \let\f@family\default@k@family % !!!
527 \begingroup
528 \try@load@fontshape
529 \endgroup
530 \fi \fi
531 \fi
532 \@font@warning{Font shape ‘\expandafter\string\reserved@a’
533 \expandafter\@gobble\string\@undefined\MessageBreak
534 using ‘\curr@fontshape’ instead\@wrong@font@char}%
535 \global\let\last@fontshape\reserved@a
536 \gdef\@defaultsubs{%
537 \@font@warning{Some font shapes were not available, defaults
538 substituted.\@gobbletwo}}%
539 \global\expandafter\expandafter\expandafter\let
540 \expandafter\reserved@a
541 \csname\curr@fontshape\endcsname
542 \xdef\font@name{%
543 \csname\curr@fontshape/\f@size\endcsname}%
544 \pickup@font}

```

そして、エンコーディングに応じて欧文用と和文用を使い分けます。

```

545 \def\wrong@fontshape{%
546 \edef\tmp@item{{\f@encoding}}%
547 \expandafter\expandafter\expandafter
548 \inlist@\expandafter\tmp@item\expandafter{\kenc@list}%
549 \ifin@
550 \wrong@ja@fontshape
551 \else
552 \wrong@al@fontshape
553 \fi
554 }
555 \</plcore | latexrelease>
556 \<latexrelease>\plEndIncludeInRelease
557 \<latexrelease>\plIncludeInRelease{2015/01/01}{\wrong@fontshape}
558 \<latexrelease> {LaTeX2e original (2015)}%
559 \<latexrelease>\def\wrong@fontshape{%
560 \<latexrelease> \csname D@f@encoding\endcsname % install defaults if in math

```

```

561 <platexrelease> \edef\reserved@a{\csname\curr@fontshape\endcsname}%
562 <platexrelease> \ifx\last@fontshape\reserved@a
563 <platexrelease> \errmessage{Corrupted NFSS tables}%
564 <platexrelease> \error@fontshape
565 <platexrelease> \else
566 <platexrelease> \let\f@shape\default@shape
567 <platexrelease> \expandafter\ifx\csname\curr@fontshape\endcsname\relax
568 <platexrelease> \let\f@series\default@series
569 <platexrelease> \expandafter
570 <platexrelease> \ifx\csname\curr@fontshape\endcsname\relax
571 <platexrelease> \let\f@family\default@family
572 <platexrelease> \begingroup
573 <platexrelease> \try@load@fontshape
574 <platexrelease> \endgroup
575 <platexrelease> \fi \fi
576 <platexrelease> \fi
577 <platexrelease> \@font@warning{Font shape '\expandafter\string\reserved@a'
578 <platexrelease> \expandafter\@gobble\string\@undefined\MessageBreak
579 <platexrelease> using '\curr@fontshape' instead\@wrong@font@char}%
580 <platexrelease> \global\let\last@fontshape\reserved@a
581 <platexrelease> \gdef\@defaultsubs{%
582 <platexrelease> \@font@warning{Some font shapes were not available, defaults
583 <platexrelease> substituted.\@gobbletwo}}%
584 <platexrelease> \global\expandafter\expandafter\expandafter\let
585 <platexrelease> \expandafter\reserved@a
586 <platexrelease> \csname\curr@fontshape\endcsname
587 <platexrelease> \xdef\font@name{%
588 <platexrelease> \csname\curr@fontshape/\f@size\endcsname}%
589 <platexrelease> \pickup@font}
590 <platexrelease> \let\wrong@a\@fontshape\@undefined
591 <platexrelease> \let\wrong@j\@fontshape\@undefined
592 <platexrelease> \plEndIncludeInRelease
593 <platexrelease> \plIncludeInRelease{0000/00/00}{\wrong@fontshape}
594 <platexrelease> {LaTeX2e original (old)}%
595 <platexrelease> \def\wrong@fontshape{%
596 <platexrelease> \csname D@f@encoding\endcsname
597 <platexrelease> \edef\reserved@a{\csname\curr@fontshape\endcsname}%
598 <platexrelease> \ifx\last@fontshape\reserved@a
599 <platexrelease> \errmessage{Corrupted NFSS tables}%
600 <platexrelease> \error@fontshape
601 <platexrelease> \else
602 <platexrelease> \let\f@shape\default@shape
603 <platexrelease> \expandafter\ifx\csname\curr@fontshape\endcsname\relax
604 <platexrelease> \let\f@series\default@series
605 <platexrelease> \expandafter
606 <platexrelease> \ifx\csname\curr@fontshape\endcsname\relax
607 <platexrelease> \let\f@family\default@family
608 <platexrelease> \fi \fi
609 <platexrelease> \fi
610 <platexrelease> \@font@warning{Font shape
611 <platexrelease> '\expandafter\string\reserved@a'
612 <platexrelease> \expandafter\@gobble\string\@undefined
613 <platexrelease> \MessageBreak
614 <platexrelease> using '\curr@fontshape' instead\@wrong@font@char}%

```

```

615 <platexrelease> \global\let\last@fontshape\reserved@a
616 <platexrelease> \gdef\@defaultsubs{%
617 <platexrelease> \font@warning{Some font shapes were not available,
618 <platexrelease> defaults substituted.\@gobbletwo}}%
619 <platexrelease> \global\expandafter\expandafter\expandafter\let
620 <platexrelease> \expandafter\reserved@a
621 <platexrelease> \csname\curr@fontshape\endcsname
622 <platexrelease> \xdef\font@name{%
623 <platexrelease> \csname\curr@fontshape/\f@size\endcsname}%
624 <platexrelease> \pickup@font}
625 <platexrelease>\let\wrong@al@fontshape\@undefined
626 <platexrelease>\let\wrong@ja@fontshape\@undefined
627 <platexrelease>\plEndIncludeInRelease
628 <*plcore>

```

`\DeclareFixedFont` フォント名を宣言するコマンドです。エンコード／ファミリ／シリーズ／シェイプ／サイズの 5 つの属性を一度に切り替えるためのコマンドを定義できます。

```

629 \def\DeclareFixedFont#1#2#3#4#5#6{%
630   \begingroup
631     \let\afont\font
632     \math@fontsfalse
633     \every@math@size{}%
634     \fontsize{#6}\z@
635     \edef\tmp@item{{#2}}%
636     \expandafter\expandafter\expandafter
637     \inlist@\expandafter\tmp@item\expandafter{\kyenc@list}%
638     \ifin@
639       \usekanji{#2}{#3}{#4}{#5}%
640       \let\font\jfont
641     \else
642       \expandafter\expandafter\expandafter
643       \inlist@\expandafter\tmp@item\expandafter{\ktenc@list}%
644       \ifin@
645         \usekanji{#2}{#3}{#4}{#5}%
646         \let\font\tfont
647       \else
648         \useroman{#2}{#3}{#4}{#5}%
649         \let\font\afont
650       \fi
651     \fi
652     \global\expandafter\let\expandafter#1\the\font
653     \let\font\afont
654   \endgroup
655 }

```

`\do@subst@correction \font` は欧文フォントを返すため、 \LaTeX の元の `\do@subst@correction` は和文

`\pltx@do@subst@correction@al` フォントに対して使えませんので、和文に対応させます¹。

`\pltx@do@subst@correction@yoko` ¹ \LaTeX 2_ε 2020-04-12 で対応。元のアスキー版の文書にも第 7.1.3 節で `\do@subst@correction` を日本語対応させた旨が書かれていましたが、実際にはこの命令は

- `\selectfont` 内の `\pickup@font` から呼ばれる場合
- `\getanddefine@fonts` 内の `\pickup@font` から呼ばれる場合

の 2 通りがあるようです。前者は `\let\font\jfont` によって対処できていましたが、後者は未対策だっ

オリジナルの L^AT_EX の定義は、欧文用として使います。

```
656 </plcore>
657 <latexrelease>\plIncludeInRelease{2020/04/12}{\do@subst@correction}
658 <latexrelease> {Japanese font substitution}%
659 <*plcore | latexrelease>
660 \def\pltx@do@subst@correction@al{%
661     \xdef\subst@correction{%
662         \font@name
663         \global\expandafter\font
664         \csname \curr@fontshape/\f@size\endcsname
665         \noexpand\fontname\font
666         \relax}%
667     \aftergroup\subst@correction
668 }
```

和文横組用と和文縦組用の定義では、それぞれ `\jfont` と `\tfont` を使います。

```
669 \def\pltx@do@subst@correction@yoko{%
670     \xdef\subst@correction{%
671         \font@name
672         \global\expandafter\jfont
673         \csname \curr@fontshape/\f@size\endcsname
674         \noexpand\fontname\jfont
675         \relax}%
676     \aftergroup\subst@correction
677 }
678 \def\pltx@do@subst@correction@tate{%
679     \xdef\subst@correction{%
680         \font@name
681         \global\expandafter\tfont
682         \csname \curr@fontshape/\f@size\endcsname
683         \noexpand\fontname\tfont
684         \relax}%
685     \aftergroup\subst@correction
686 }
```

そして、エンコーディングに応じて 3 つの命令を使い分けます。

```
687 \def\do@subst@correction{%
688     \edef\tmp@item{{\f@encoding}}%
689     \expandafter\expandafter\expandafter
690     \inlist@\expandafter\tmp@item\expandafter{\kyenc@list}%
691     \ifin@\pltx@do@subst@correction@yoko
692     \else
693         \expandafter\expandafter\expandafter
694         \inlist@\expandafter\tmp@item\expandafter{\ktenc@list}%
695         \ifin@\pltx@do@subst@correction@tate\else
696             \pltx@do@subst@correction@al
697         \fi
698     \fi
699 }
700 </plcore | latexrelease>
701 <latexrelease>\plEndIncludeInRelease
702 <latexrelease>\plIncludeInRelease{0000/00/00}{\do@subst@correction}
```

ため、例えば和文数式フォントを定義した状態で `bm` パッケージを使った場合に問題が起きていました (参考: `texjporg/jsclasses#53`)。

```

703 <platexrelease>                                {LaTeX2e original}%
704 <platexrelease>\def\do@subst@correction{%
705 <platexrelease>    \xdef\subst@correction{%
706 <platexrelease>        \font@name
707 <platexrelease>        \global\expandafter\font
708 <platexrelease>        \csname \curr@fontshape/\f@size\endcsname
709 <platexrelease>        \noexpand\fontname\font
710 <platexrelease>        \relax}%
711 <platexrelease>    \aftergroup\subst@correction
712 <platexrelease>}
713 <platexrelease>\let\pltx@do@subst@correction@al@undefined
714 <platexrelease>\let\pltx@do@subst@correction@yoko@undefined
715 <platexrelease>\let\pltx@do@subst@correction@tate@undefined
716 <platexrelease>\plEndIncludeInRelease
717 <*plcore>

```

7.2.3 数式用フォント

`\reDeclareMathAlphabet` 数式モード内で、数式文字用の和欧文フォントを同時に切り替えるコマンドです。

p_lTEX 2_ε には、本来の動作モードと 2.09 互換モードの二つがあり、両モードで数式文字を変更するコマンドや動作が異なります。本来の動作モードでは、`\mathrm{...}` のように `\math??` に引数を指定して使います。このときは引数にだけ影響します。2.09 互換モードでは、`\rm` のような二文字コマンドを使います。このコマンドには引数を取らず、書体はグルーピングの範囲で反映されます。二文字コマンドは、ネイティブモードでも使えるようになっていて、動作も 2.09 互換モードのコマンドと同じです。

しかし、内部的には `\math??` という一つのコマンドがすべての動作を受け持ち、`\math??` コマンドや `\??` コマンドから呼び出された状態に応じて、動作を変えています。したがって、欧文フォントと和文フォントの両方を一度に変更する、数式文字変更コマンドを作るとき、それぞれの状態に合った動作で動くようにフォント切り替えコマンドを実行させる必要があります。

使い方

usage: `\reDeclareMathAlphabet{\mathAA}{\mathBB}{\mathCC}`

欧文・和文両用の数式文字変更コマンド `\mathAA` を (再) 定義します。欧文用のコマンド `\mathBB` と、和文用の `\mathCC` を (p)TEX 標準の方法で定義しておいた後、上のように記述します。なお、`{\mathBB}{\mathCC}` の部分については `{\@mathBB}{\@mathCC}` のように `@` をつけた記述をしてもかまいません (互換性のため)。上のような命令を発行すると、`\mathAA` が、欧文に対しては `\mathBB`、和文に対しては `\mathCC` の意味を持つようになります。通常は、`\reDeclareMathAlphabet{\mathrm}{\mathrm}{\mathmc}` のように `AA=BB` として用います。また、`\mathrm` は L^AT_EX kernel において標準のコマンドとして既に定義されているので、この場合は `\mathrm` の再定義となります。native mode での `\rm` のような two letter command (old font command) に対しても同様なことが引き起こります。つまり、数式モードにおいて、新たな `\rm` は、L^AT_EX original

の `\rm` と `\mc` (正確に言えば `\mathrm` と `\mathmc` であるが) の意味を合わせ持つようになります。

補足

- `\mathAA` を再定義する他の命令 (`\DeclareSymbolFontAlphabet` を用いるパッケージの使用等) との衝突を避けるためには、`\AtBeginDocument` を併用するなどして展開位置の制御を行ってください。
- テキストモード時のエラー表示用に `\mathBB` のみを用いることを除いて、`\mathBB` と `\mathCC` の順は実際には意味を持ちません。和文、欧文の順に定義しても問題はありません。
- 第 2,3 引き数には `{\@mathBB}{\@mathCC}` のように `@` をつけた記述も行えます。ただし、形式は統一してください。判断は第 2 引き数で行っているため、`{\@mathBB}{\mathCC}` のような記述ではうまく動作しません。また、`\makeatletter` な状態で `{\@mathBB }{\@mathCC }` のような `@` と余分なスペースをつけた場合には無限ループを引き起こすことがあります。このような記述は避けるようにして下さい。
- `\reDeclareMathAlphabet` を実行する際には、`\mathBB`, `\mathCC` が定義されている必要はありません。実際に `\mathAA` を用いる際にはこれらの `\mathBB`, `\mathCC` が (p)LaTeX 標準の方法で定義されている必要があります。
- 他の部分で `\mathAA` を全く定義しない場合を除き、`\mathAA` は `\reDeclareMathAlphabet` を実行する以前で (p)LaTeX 標準の方法で定義されている必要があります (`\mathrm` や `\mathbf` の標準的なコマンドは、LaTeX kernel で既に定義されています)。 `\DeclareMathAlphabet` の場合には、`\reDeclareMathAlphabet` よりも前で 1 度 `\mathAA` を定義してあれば、`\reDeclareMathAlphabet` の後ろで再度 `\DeclareMathAlphabet` を用いて `\mathAA` の内部の定義内容を変更することには問題ありません。 `\DeclareSymbolFontAlphabet` の場合、再定義においても `\mathAA` が直接定義されるので、`\mathAA` に対する最後の `\DeclareSymbolFontAlphabet` のさらに後で `\reDeclareMathAlphabet` を実行しなければ有効とはなりません。
- `\documentstyle` の互換モードの場合、`\rm` 等の two letter command (old font command) は、`\reDeclareMathAlphabet` とは関連することのない別個のコマンドとして定義されます。従って、この場合には `\reDeclareMathAlphabet` を用いても `\rm` 等は数式モードにおいて欧文・和文両用のものとはなりません。

```

718 \def\reDeclareMathAlphabet#1#2#3{%
719   \edef#1{\noexpand\protect\expandafter\noexpand\csname%
720     \expandafter@gobble\string#1\space\space\endcsname}%
721   \edef\@tempa{\expandafter@gobble\string#2}%
722   \edef\@tempb{\expandafter@gobble\string#3}%
723   \edef\@tempc{\string @\expandafter@gobbletwo\string#2}%

```

```

724 \ifx\@tempc\@tempa%
725 \edef\@tempa{\expandafter\@gobbletwo\string#2}%
726 \edef\@tempb{\expandafter\@gobbletwo\string#3}%
727 \fi
728 \expandafter\edef\csname\expandafter\@gobble\string#1\space\space\endcsname%
729 {\noexpand\DualLang@mathalph@bet%
730 {\expandafter\noexpand\csname\@tempa\space\endcsname}%
731 {\expandafter\noexpand\csname\@tempb\space\endcsname}%
732 }%
733 }
734 \@onlypreamble\reDeclareMathAlphabet
735 \def\DualLang@mathalph@bet#1#2{%
736 \relax\ifmmode
737 \ifx\math@bgroup\bgroup% 2e normal style (\mathrm{...})
738 \bgroup\let\DualLang@Mfontsw\DLMfontsw@standard
739 \else
740 \ifx\math@bgroup\relax% 2e two letter style (\rm->\mathrm)
741 \let\DualLang@Mfontsw\DLMfontsw@oldstyle
742 \else
743 \ifx\math@bgroup\@empty% 2.09 oldfont style ({\mathrm ...})
744 \let\DualLang@Mfontsw\DLMfontsw@oldfont
745 \else% panic! assume 2e normal style
746 \bgroup\let\DualLang@Mfontsw\DLMfontsw@standard
747 \fi
748 \fi
749 \fi
750 \else
751 \let\DualLang@Mfontsw\@firstoftwo
752 \fi
753 \DualLang@Mfontsw{#1}{#2}%
754 }
755 \def\DLMfontsw@standard#1#2#3{#1{#2{#3}}\egroup}
756 \def\DLMfontsw@oldstyle#1#2{#1\relax\@fontswitch\relax{#2}}
757 \def\DLMfontsw@oldfont#1#2{#1\relax#2\relax}

```

7.2.4 従属書体の宣言

`\DeclareRelationFont` 和文書体に対する従属書体を宣言するコマンドです。従属書体とは、ある和文書体とペアになる欧文書体のことです。主に多書体パッケージ `skfonts` を用いるための仕組みです。

`\DeclareRelationFont` コマンドの最初の 4 つの引数の組が和文書体の属性、その後の 4 つの引数の組が従属書体の属性です。

```

\DeclareRelationFont{JY1}{mc}{m}{n}{OT1}{cmr}{m}{n}
\DeclareRelationFont{JY1}{gt}{m}{n}{OT1}{cmr}{bx}{n}

```

上記の例は、明朝体の従属書体としてコンピュータモダンローマン、ゴシック体の従属書体としてコンピュータモダンボールドを宣言しています。カレント和文書体が `\JY1/mc/m/n` となると、自動的に欧文書体が `\OT1/cmr/m/n` になります。また、和文書体が `\JY1/gt/m/n` になったときは、欧文書体が `\OT1/cmr/bx/n` になります。

和文書体のシェイプ指定を省略するとエンコード／ファミリ／シリーズの組合せ

で従属書体が使われます。このときは、`\selectfont` が呼び出された時点でのシェイプ (`\f@shape`) の値が使われます。

`\DeclareRelationFont` の設定値はグローバルに有効です。`\SetRelationFont` の設定値はローカルに有効です。フォント定義ファイルで宣言をする場合は、`\DeclareRelationFont` を使ってください。

```

758 \def\all@shape{all}%
759 \def\DeclareRelationFont#1#2#3#4#5#6#7#8{%
760   \def\rel@shape{#4}%
761   \ifx\rel@shape\@empty
762     \global
763     \expandafter\def\csname rel@#1/#2/#3/all\endcsname{%
764       \romanencoding{#5}\romanfamily{#6}%
765       \romanseries{#7}}%
766   \else
767     \global
768     \expandafter\def\csname rel@#1/#2/#3/#4\endcsname{%
769       \romanencoding{#5}\romanfamily{#6}%
770       \romanseries{#7}\romanshape{#8}}%
771   \fi
772 }
773 \def\SetRelationFont#1#2#3#4#5#6#7#8{%
774   \def\rel@shape{#4}%
775   \ifx\rel@shape\@empty
776     \expandafter\def\csname rel@#1/#2/#3/all\endcsname{%
777       \romanencoding{#5}\romanfamily{#6}%
778       \romanseries{#7}}%
779   \else
780     \expandafter\def\csname rel@#1/#2/#3/#4\endcsname{%
781       \romanencoding{#5}\romanfamily{#6}%
782       \romanseries{#7}\romanshape{#8}}%
783   \fi
784 }

```

`\if@knjcmd` `\if@knjcmd` は欧文書体を従属書体にするかどうかのフラグです。このフラグが真になると、欧文書体に従属書体が使われます。

```

785 \newif\if@knjcmd

```

`\userelfont` `\if@knjcmd` フラグは `\userelfont` コマンドによって、真となります。そして `\selectfont` 実行後には偽に初期化されます。

```

786 \<plcore>
787 \<latexrelease>\plIncludeInRelease{2019/10/01}{\userelfont}
788 \<latexrelease>          {Make robust}%
789 \<*plcore | latexrelease>
790 \DeclareRobustCommand\userelfont{\@knjcmdtrue}
791 \<plcore | latexrelease>
792 \<latexrelease>\plEndIncludeInRelease
793 \<latexrelease>\plIncludeInRelease{0000/00/00}{\userelfont}
794 \<latexrelease>          {ASCII Corporation original}%
795 \<latexrelease>\def\userelfont{\@knjcmdtrue}
796 \<latexrelease>\expandafter \let \csname userelfont \endcsname \@undefined
797 \<latexrelease>\plEndIncludeInRelease

```

798 <*plcore>

7.2.5 フォントの選択

\selectfont \selectfont のオリジナルからの変更部分は、次の 3 点です。

- 和文書体を変更する部分
- 従属書体に変更する部分
- 和欧文のベースラインを調整する部分

```
799 </plcore>
800 <latexrelease | trace>\plIncludeInRelease{2020/04/12}{\selectfont}
801 <latexrelease | trace> {Check \KanjiEncodingPair}%
802 <*plcore | latexrelease | trace>
803 \ifx\delayed@f@adjustment\undefined % --- for <= 2020-10-01 BEGIN
804 %%
805 \DeclareRobustCommand\selectfont{%
806   \let\tmp@error@fontshape\error@fontshape
807   \let\error@fontshape\error@kfontshape
808   \edef\tmp@item{{\k@encoding}}}%
809   \expandafter\expandafter\expandafter
810   \inlist@\expandafter\tmp@item\expandafter{\kyenc@list}%
811   \ifin@
812     \let\cy@encoding\k@encoding
813     \ensure@KanjiEncodingPair{t}%
814     \edef\ct@encoding{\csname t@enc@\k@encoding\endcsname}%
815   \else
816     \expandafter\expandafter\expandafter
817     \inlist@\expandafter\tmp@item\expandafter{\ktenc@list}%
818     \ifin@
819       \let\ct@encoding\k@encoding
820       \ensure@KanjiEncodingPair{y}%
821       \edef\cy@encoding{\csname y@enc@\k@encoding\endcsname}%
822     \else
823       \latex@error{KANJI Encoding scheme ‘\k@encoding’ unknown}\@eha
824     \fi
825   \fi
826   \let\font\tfont
827   \let\k@encoding\ct@encoding
828   \xdef\font@name{\csname curr@kfontshape/\f@size\endcsname}%
829   \pickup@font
830   \font@name
831   \let\font\jfont
832   \let\k@encoding\cy@encoding
833   \xdef\font@name{\csname curr@kfontshape/\f@size\endcsname}%
834   \pickup@font
835   \font@name
836   \expandafter\def\expandafter\k@encoding\tmp@item
837   \kenc@update
838   \let\error@fontshape\tmp@error@fontshape
839   \if@knjcmd \@knjcmdfalse
```

```

840 \expandafter\ifx
841 \csname rel@k@encoding/k@family/k@series/k@shape\endcsname\relax
842 \expandafter\ifx
843 \csname rel@k@encoding/k@family/k@series/all\endcsname\relax
844 \else
845 \csname rel@k@encoding/k@family/k@series/all\endcsname
846 \fi
847 \else
848 \csname rel@k@encoding/k@family/k@series/k@shape\endcsname
849 \fi
850 \fi
851 \let\font\afont
852 \xdef\font@name{\csname\curr@fontshape/f@size\endcsname}%
853 \pickup@font
854 \font@name
855 \trace \ifnum \tracingfonts>\tw@
856 \trace \@font@info{Roman:Switching to \font@name}\fi
857 \enc@update
858 \ifx\f@linespread\baselinestretch \else
859 \set@fontsize\baselinestretch\f@size\f@baselineskip
860 \fi
861 \size@update}
862 %%
863 \else % --- for <= 2020-10-01 END & for >= 2021-06-01 BEGIN
864 %%
865 \DeclareRobustCommand\selectfont{%

```

最初に和文の処理を完了させてから、欧文（従属欧文かもしれません）の処理に入る必要があります。それぞれで「遅らせていたシリーズ・シェイプの値更新→フォントの切り替え」を行います。

和文：遅らせていたシリーズ・シェイプの値更新。

```

866 % !! sync with ltfsstrc.dtx 2021/04/26 v3.0o BEGIN
867 \ifx\delayed@k@adjustment\@empty
868 \else
869 \let\k@shape@saved\k@shape
870 \let\k@series@saved\k@series
871 \delayed@k@adjustment
872 \begingroup\let\f@encoding\k@encoding\let\f@family\k@family
873 \maybe@load@fontshape@endgroup
874 \ifcsname \k@encoding/k@family/k@series/k@shape \endcsname
875 \else
876 \let\k@shape\k@shape@saved
877 \let\k@series\k@series@saved
878 \let\delayed@merge@kanji@shape\merge@kanji@shape
879 \let\delayed@merge@kanji@series\merge@kanji@series
880 \delayed@k@adjustment
881 \let\delayed@merge@kanji@shape\merge@kanji@shape@without@substitution
882 \let\delayed@merge@kanji@series\merge@kanji@series@without@substitution
883 \fi
884 \let\delayed@k@adjustment\@empty
885 \fi
886 \@forced@series@kanjifalse
887 % !! sync with ltfsstrc.dtx 2021/04/26 v3.0o END

```

和文：フォントの切り替え。

```
888 \let\tmp@error@fontshape\error@fontshape
889 \let\error@fontshape\error@kfontshape
890 \edef\tmp@item{{\k@encoding}}%
891 \expandafter\expandafter\expandafter
892 \inlist@\expandafter\tmp@item\expandafter{\kyenc@list}%
893 \ifin@
894 \let\cy@encoding\k@encoding
895 \ensure@KanjiEncodingPair{t}%
896 \edef\ct@encoding{\csname t@enc@\k@encoding\endcsname}%
897 \else
898 \expandafter\expandafter\expandafter
899 \inlist@\expandafter\tmp@item\expandafter{\ktenc@list}%
900 \ifin@
901 \let\ct@encoding\k@encoding
902 \ensure@KanjiEncodingPair{y}%
903 \edef\cy@encoding{\csname y@enc@\k@encoding\endcsname}%
904 \else
905 \@latex@error{KANJI Encoding scheme ‘\k@encoding’ unknown}\@eha
906 \fi
907 \fi
908 \let\font\tfont
909 \let\k@encoding\ct@encoding
910 \xdef\font@name{\csname\curr@kfontshape/\f@size\endcsname}%
911 \pickup@font
912 \font@name
913 \let\font\jfont
914 \let\k@encoding\cy@encoding
915 \xdef\font@name{\csname\curr@kfontshape/\f@size\endcsname}%
916 \pickup@font
917 \font@name
918 \expandafter\def\expandafter\k@encoding\tmp@item
919 \kenc@update
920 \let\error@fontshape\tmp@error@fontshape
```

以上で、和文の処理が完了。

従属欧文を使う（\if@knjcmd が真の）場合は、和文書体が確定したこの段階で「関連付けされた欧文書体」に値を更新します。 \rel@../../../.. で \romanseries と \romanshape が実行される（すなわち \delayed@f@adjustment に追加が入る）ことに注意してください。なお、\if@knjcmd フラグは \userelfont コマンドによって真となり、以下で再び偽にリセットされます。

```
921 \if@knjcmd \@knjcmdfalse
922 \expandafter\ifx
923 \csname rel@\k@encoding/\k@family/\k@series/\k@shape\endcsname\relax
924 \expandafter\ifx
925 \csname rel@\k@encoding/\k@family/\k@series/all\endcsname\relax
926 \else
927 \csname rel@\k@encoding/\k@family/\k@series/all\endcsname
928 \fi
929 \else
930 \csname rel@\k@encoding/\k@family/\k@series/\k@shape\endcsname
931 \fi
```


932 \fi

欧文：遅らせていたシリーズ・シェイプの値更新。

```
933 % !! sync with ltfsstrc.dtx 2021/04/26 v3.0o BEGIN
934 \ifx\delayed@f@adjustment\@empty
935 \else
936   \let\f@shape@saved\f@shape
937   \let\f@series@saved\f@series
938   \delayed@f@adjustment
939   \maybe@load@fontshape
940   \ifcsname \f@encoding/\f@family/\f@series/\f@shape \endcsname
941   \else
942     \let\f@shape\f@shape@saved
943     \let\f@series\f@series@saved
944     \let\delayed@merge@font@shape\merge@font@shape
945     \let\delayed@merge@font@series\merge@font@series
946     \delayed@f@adjustment
947     \let\delayed@merge@font@shape\merge@font@shape@without@substitution
948     \let\delayed@merge@font@series\merge@font@series@without@substitution
949   \fi
950   \let\delayed@f@adjustment\@empty
951 \fi
952 \@forced@seriesfalse
953 % !! sync with ltfsstrc.dtx 2021/04/26 v3.0o END
```

欧文：フォントの切り替え。

```
954 \let\font\afont
955 \xdef\font@name{\csname\curr@fontshape/\f@size\endcsname}%
956 \pickup@font
957 \font@name
958 \UseHook{selectfont}% since LaTeX2e 2021-06-01
959 \enc@update
```

最後に、サイズが変更されていれば、ベースラインの調整などを行ないます。英語版の `\selectfont` では最初に行なっていますが、`pLATEX 2ε` ではベースラインシフトの調整をするために、書体を確定しなければならないため、一番最後に行ないます

```
960 \ifx\f@linespread\baselinestretch \else
961   \set@fontsize\baselinestretch\f@size\f@baselineskip
962 \fi
963 \size@update}
964 %%
965 \fi % --- for >= 2021-06-01 END
966 </plcore | platexrelease | trace>
967 <platexrelease | trace>\plEndIncludeInRelease
968 <platexrelease | trace>\plIncludeInRelease{0000/00/00}{\selectfont}
969 <platexrelease | trace> {ASCII Corporation original}%
970 <platexrelease | trace>\DeclareRobustCommand\selectfont{%
971 <platexrelease | trace> \let\tmp@error@fontshape\error@fontshape
972 <platexrelease | trace> \let\error@fontshape\error@kfontshape
973 <platexrelease | trace> \edef\tmp@item{{\k@encoding}}%
974 <platexrelease | trace> \expandafter\expandafter\expandafter
975 <platexrelease | trace> \inlist@ \expandafter\tmp@item\expandafter{\kyenc@list}%
976 <platexrelease | trace> \ifin@
977 <platexrelease | trace> \let\cy@encoding\k@encoding
```

```

978 <platexrelease|trace> \edef\ct@encoding{\csname t@enc@\k@encoding\endcsname}%
979 <platexrelease|trace> \else
980 <platexrelease|trace> \expandafter\expandafter\expandafter
981 <platexrelease|trace> \inlist@\expandafter\tmp@item\expandafter{\ktenc@list}%
982 <platexrelease|trace> \ifin@
983 <platexrelease|trace> \let\ct@encoding\k@encoding
984 <platexrelease|trace> \edef\cy@encoding{\csname y@enc@\k@encoding\endcsname}%
985 <platexrelease|trace> \else
986 <platexrelease|trace> \@latex@error{KANJI Encoding scheme ‘\k@encoding’ unknown}\@eha
987 <platexrelease|trace> \fi
988 <platexrelease|trace> \fi
989 <platexrelease|trace> \let\font\tfont
990 <platexrelease|trace> \let\k@encoding\ct@encoding
991 <platexrelease|trace> \xdef\font@name{\csname\curr@kfontshape/\f@size\endcsname}%
992 <platexrelease|trace> \pickup@font
993 <platexrelease|trace> \font@name
994 <platexrelease|trace> \let\font\jfont
995 <platexrelease|trace> \let\k@encoding\cy@encoding
996 <platexrelease|trace> \xdef\font@name{\csname\curr@kfontshape/\f@size\endcsname}%
997 <platexrelease|trace> \pickup@font
998 <platexrelease|trace> \font@name
999 <platexrelease|trace> \expandafter\def\expandafter\k@encoding\tmp@item
1000 <platexrelease|trace> \kenc@update
1001 <platexrelease|trace> \let\error@fontshape\tmp@error@fontshape
1002 <platexrelease|trace> \if@knjcmd \@knjcmdfalse
1003 <platexrelease|trace> \expandafter\ifx
1004 <platexrelease|trace> \csname rel@\k@encoding/\k@family/\k@series/\k@shape\endcsname\relax
1005 <platexrelease|trace> \expandafter\ifx
1006 <platexrelease|trace> \csname rel@\k@encoding/\k@family/\k@series/all\endcsname\relax
1007 <platexrelease|trace> \else
1008 <platexrelease|trace> \csname rel@\k@encoding/\k@family/\k@series/all\endcsname
1009 <platexrelease|trace> \fi
1010 <platexrelease|trace> \else
1011 <platexrelease|trace> \csname rel@\k@encoding/\k@family/\k@series/\k@shape\endcsname
1012 <platexrelease|trace> \fi
1013 <platexrelease|trace> \fi
1014 <platexrelease|trace> \let\font\afont
1015 <platexrelease|trace> \xdef\font@name{\csname\curr@fontshape/\f@size\endcsname}%
1016 <platexrelease|trace> \pickup@font
1017 <platexrelease|trace> \font@name
1018 <*trace>
1019 <platexrelease|trace> \ifnum \tracingfonts>\tw@
1020 <platexrelease|trace> \@font@info{Roman:Switching to \font@name}\fi
1021 </trace>
1022 <platexrelease|trace> \enc@update
1023 <platexrelease|trace> \ifx\f@linespread\baselinestretch \else
1024 <platexrelease|trace> \set@fontsize\baselinestretch\f@size\f@baselineskip
1025 <platexrelease|trace> \fi
1026 <platexrelease|trace> \size@update}
1027 <platexrelease|trace> \plEndIncludeInRelease
1028 <*plcore>

```

\set@fontsize \fontsize コマンドの内部形式です。ベースラインの設定と、支柱の設定を行ない

ます。

```
1029 </plcore>
1030 <latexrelease | trace>\plIncludeInRelease{2017/04/08}{\set@fontsize}
1031 <latexrelease | trace>          {Construct \ystrutbox}%
1032 <*plcore | latexrelease | trace>
1033 \def\set@fontsize#1#2#3{%
1034     \@defaultunits\@tempdimb#2pt\relax\@nnil
1035     \edef\f@size{\strip@pt\@tempdimb}%
1036     \@defaultunits\@tempskipa#3pt\relax\@nnil
1037     \edef\f@baselineskip{\the\@tempskipa}%
1038     \edef\f@linespread{#1}%
1039     \let\baselinestretch\f@linespread
1040     \def\size@update{%
1041         \baselineskip\f@baselineskip\relax
1042         \baselineskip\f@linespread\baselineskip
1043         \normalbaselineskip\baselineskip
```

ここで、ベースラインシフトの調整と支柱を組み立てます。

```
1044     \adjustbaseline
1045     \setbox\ystrutbox\hbox{\yoko
1046         \vrule\@width\z@
1047             \@height.7\baselineskip \@depth.3\baselineskip}%
1048     \setbox\tstrutbox\hbox{\tate
1049         \vrule\@width\z@
1050             \@height.5\baselineskip \@depth.5\baselineskip}%
1051     \setbox\zstrutbox\hbox{\tate
1052         \vrule\@width\z@
1053             \@height.7\baselineskip \@depth.3\baselineskip}%
```

フォントサイズとベースラインに関する診断情報を出力します。

```
1054 <*trace>
1055     \ifnum \tracingfonts>\tw@
1056     \ifx\f@linespread\@empty
1057         \let\reserved@a\@empty
1058     \else
1059         \def\reserved@a{\f@linespread x}%
1060     \fi
1061     \@font@info{Changing size to\space
1062         \f@size/\reserved@a \f@baselineskip}%
1063     \aftergroup\type@restoreinfo
1064     \fi
1065 </trace>
1066     \let\size@update\relax}}
1067 </plcore | latexrelease | trace>
1068 <latexrelease | trace>\plEndIncludeInRelease
1069 <latexrelease | trace>\plIncludeInRelease{0000/00/00}{\set@fontsize}
1070 <latexrelease | trace>          {ASCII Corporation original}%
1071 <latexrelease | trace>\def\set@fontsize#1#2#3{%
1072 <latexrelease | trace>     \@defaultunits\@tempdimb#2pt\relax\@nnil
1073 <latexrelease | trace>     \edef\f@size{\strip@pt\@tempdimb}%
1074 <latexrelease | trace>     \@defaultunits\@tempskipa#3pt\relax\@nnil
1075 <latexrelease | trace>     \edef\f@baselineskip{\the\@tempskipa}%
1076 <latexrelease | trace>     \edef\f@linespread{#1}%
1077 <latexrelease | trace>     \let\baselinestretch\f@linespread
```

```

1078 <platexrelease | trace> \def\size@update{%
1079 <platexrelease | trace> \baselineskip\f@baselineskip\relax
1080 <platexrelease | trace> \baselineskip\f@linespread\baselineskip
1081 <platexrelease | trace> \normalbaselineskip\baselineskip
1082 <platexrelease | trace> \adjustbaseline
1083 <platexrelease | trace> \setbox\strutbox\hbox{\yoko
1084 <platexrelease | trace> \vrule\@width\z@
1085 <platexrelease | trace> \@height.7\baselineskip \@depth.3\baselineskip}%
1086 <platexrelease | trace> \setbox\tstrutbox\hbox{\tate
1087 <platexrelease | trace> \vrule\@width\z@
1088 <platexrelease | trace> \@height.5\baselineskip \@depth.5\baselineskip}%
1089 <platexrelease | trace> \setbox\zstrutbox\hbox{\tate
1090 <platexrelease | trace> \vrule\@width\z@
1091 <platexrelease | trace> \@height.7\baselineskip \@depth.3\baselineskip}%
1092 <*trace>
1093 <platexrelease | trace> \ifnum \tracingfonts>\tw@
1094 <platexrelease | trace> \ifx\f@linespread\@empty
1095 <platexrelease | trace> \let\reserved@a\@empty
1096 <platexrelease | trace> \else
1097 <platexrelease | trace> \def\reserved@a{\f@linespread x}%
1098 <platexrelease | trace> \fi
1099 <platexrelease | trace> \@font@info{Changing size to\space
1100 <platexrelease | trace> \f@size/\reserved@a \f@baselineskip}%
1101 <platexrelease | trace> \aftergroup\type@restoreinfo
1102 <platexrelease | trace> \fi
1103 </trace>
1104 <platexrelease | trace> \let\size@update\relax}}
1105 <platexrelease | trace> \plEndIncludeInRelease
1106 <*plcore>

```

\adjustbaseline 現在の和文フォントの空白（EUC コード 0xA1A1）の中央に現在の欧文フォントの“/”の中央がくるようにベースラインシフトを設定します。

当初はまずベースラインシフト量をゼロにしていたましたが、\tbaselineshiftを連続して変更した後に鉤括弧類を使うと余計なアキがでる問題が起こるため、\tbaselineshift をゼロクリアする処理を削除しました。

しかし、それではベースラインシフトを調整済みの欧文ボックスと比較してしまうため、計算した値が大きくなってしまいます。そこで、このボックスの中でゼロにするようにしました。また、“/”と比較していたのを“M”にしました。

全角空白（EUC コード 0xA1A1）は JFM で特殊なタイプに分類される可能性があるため、和文書体の基準を「漢」（JIS コード 0x3441）へ変更しました。

```

1107 \newbox\adjust@box
1108 \newdimen\adjust@dimen
1109 </plcore>
1110 <platexrelease | trace> \plIncludeInRelease{2019/10/01}{\adjustbaseline}
1111 <platexrelease | trace> {Make robust}%
1112 <*plcore | platexrelease | trace>
1113 \DeclareRobustCommand\adjustbaseline{%

```

和文フォントの基準値を設定します。

```

1114 \setbox\adjust@box\hbox{\char\jis"3441}%

```

```

1115 \cht\ht\adjust@box
1116 \cdp\dp\adjust@box
1117 \c wd\wd\adjust@box
1118 \cvs\normalbaselineskip
1119 \chs\c wd
1120 \cHT\cht \advance\cHT\cdp

```

基準となる欧文フォントの文字を含んだボックスを作成し、ベースラインシフト量の計算を行ないます。計算式は次のとおりです。

$$\text{ベースラインシフト量} = \frac{\{(\text{漢の深さ}) - (\text{M の深さ})\} \times (\text{漢の高さ} + \text{深さ}) - (\text{M の高さ} + \text{深さ})}{2}$$

```

1121 \iftdir
1122 \setbox\adjust@box\hbox{\tbaselineshift\z@ M}%
1123 \adjust@dimen\ht\adjust@box
1124 \advance\adjust@dimen\dp\adjust@box
1125 \advance\adjust@dimen-\cHT
1126 \divide\adjust@dimen\tw@
1127 \advance\adjust@dimen\cdp
1128 \advance\adjust@dimen-\dp\adjust@box
1129 \tbaselineshift\adjust@dimen
1130 <trace> \ifnum \tracingfonts>\tw@
1131 <trace> \typeout{baselineshift:\the\tbaselineshift}%
1132 <trace> \fi
1133 \fi}
1134 </plcore | platexrelease | trace>
1135 <platexrelease | trace>\plEndIncludeInRelease
1136 <platexrelease | trace>\plIncludeInRelease{2017/07/29}{\adjustbaseline}
1137 <platexrelease | trace> {Change zenkaku reference}%
1138 <platexrelease | trace>\def\adjustbaseline{%
1139 <platexrelease | trace> \setbox\adjust@box\hbox{\char\jis"3441}%
1140 <platexrelease | trace> \cht\ht\adjust@box
1141 <platexrelease | trace> \cdp\dp\adjust@box
1142 <platexrelease | trace> \c wd\wd\adjust@box
1143 <platexrelease | trace> \cvs\normalbaselineskip
1144 <platexrelease | trace> \chs\c wd
1145 <platexrelease | trace> \cHT\cht \advance\cHT\cdp
1146 <platexrelease | trace> \iftdir
1147 <platexrelease | trace> \setbox\adjust@box\hbox{\tbaselineshift\z@ M}%
1148 <platexrelease | trace> \adjust@dimen\ht\adjust@box
1149 <platexrelease | trace> \advance\adjust@dimen\dp\adjust@box
1150 <platexrelease | trace> \advance\adjust@dimen-\cHT
1151 <platexrelease | trace> \divide\adjust@dimen\tw@
1152 <platexrelease | trace> \advance\adjust@dimen\cdp
1153 <platexrelease | trace> \advance\adjust@dimen-\dp\adjust@box
1154 <platexrelease | trace> \tbaselineshift\adjust@dimen
1155 <*trace>
1156 <platexrelease | trace> \ifnum \tracingfonts>\tw@
1157 <platexrelease | trace> \typeout{baselineshift:\the\tbaselineshift}%

```

```

1158 <latexrelease|trace> \fi
1159 </trace>
1160 <latexrelease|trace> \fi}
1161 <latexrelease|trace>\expandafter \let \csname adjustbaseline \endcsname \@undefined
1162 <latexrelease|trace>\plEndIncludeInRelease
1163 <latexrelease|trace>\plIncludeInRelease{0000/00/00}{\adjustbaseline}
1164 <latexrelease|trace> {ASCII Corporation original}%
1165 <latexrelease|trace>\def\adjustbaseline{%
1166 <latexrelease|trace> \setbox\adjust@box\hbox{\char\eut"A1A1}%
1167 <latexrelease|trace> \cht\ht\adjust@box
1168 <latexrelease|trace> \cdp\dp\adjust@box
1169 <latexrelease|trace> \c wd\wd\adjust@box
1170 <latexrelease|trace> \cvs\normalbaselineskip
1171 <latexrelease|trace> \chs\c wd
1172 <latexrelease|trace> \cHT\cht \advance\cHT\cdp
1173 <latexrelease|trace> \iftdir
1174 <latexrelease|trace> \setbox\adjust@box\hbox{\tbaselineshift\z@ M}%
1175 <latexrelease|trace> \adjust@dimen\ht\adjust@box
1176 <latexrelease|trace> \advance\adjust@dimen\dp\adjust@box
1177 <latexrelease|trace> \advance\adjust@dimen-\cHT
1178 <latexrelease|trace> \divide\adjust@dimen\tw@
1179 <latexrelease|trace> \advance\adjust@dimen\cdp
1180 <latexrelease|trace> \advance\adjust@dimen-\dp\adjust@box
1181 <latexrelease|trace> \tbaselineshift\adjust@dimen
1182 <*trace>
1183 <latexrelease|trace> \ifnum \tracingfonts>\tw@
1184 <latexrelease|trace> \typeout{baselineshift:\the\tbaselineshift}
1185 <latexrelease|trace> \fi
1186 </trace>
1187 <latexrelease|trace> \fi}
1188 <latexrelease|trace>\expandafter \let \csname adjustbaseline \endcsname \@undefined
1189 <latexrelease|trace>\plEndIncludeInRelease
1190 <*plcore>

```

7.2.6 エンコードの指定

`\romanencoding` 書体のエンコードを指定するコマンドです。`\fontencoding` コマンドは和欧文のど
`\kanjiencoding` ちらかに影響します。`\DeclareKanjiEncoding` で指定されたエンコードは和文エ
`\fontencoding` ンコードとして、`\DeclareFontEncoding` で指定されたエンコードは欧文エンコー
ドとして認識されます。

`\kanjiencoding` と `\romanencoding` は与えられた引数が、エンコードとして登
録されているかどうかだけを確認し、それが和文か欧文かのチェックは行なってい
ません。そのため、高速に動作をしますが、`\kanjiencoding` に欧文エンコードを
指定したり、逆に `\romanencoding` に和文エンコードを指定した場合はエラーとな
ります。

```

1191 \DeclareRobustCommand\romanencoding[1]{%
1192 \expandafter\ifx\csname T@#1\endcsname\relax
1193 \latex@error{Encoding scheme ‘#1’ unknown}\@eha
1194 \else
1195 \edef\f@encoding{#1}%

```

```

1196     \ifx\cf@encoding\f@encoding
1197     \let\enc@update\relax
1198   \else
1199     \let\enc@update\@@enc@update
1200   \fi
1201 \fi
1202 }
1203 \DeclareRobustCommand\kanjiencoding[1]{%
1204   \expandafter\ifx\csname T@#1\endcsname\relax
1205     \latex@error{KANJI Encoding scheme ‘#1’ unknown}\@eha
1206   \else
1207     \edef\k@encoding{#1}%
1208     \ifx\ck@encoding\k@encoding
1209       \let\kenc@update\relax
1210     \else
1211       \let\kenc@update\@@kenc@update
1212     \fi
1213   \fi
1214 }
1215 \DeclareRobustCommand\fontencoding[1]{%
1216   \edef\tmp@item{{#1}}%
1217   \expandafter\expandafter\expandafter
1218   \inlist@\expandafter\tmp@item\expandafter{\kenc@list}%
1219   \ifin@ \kanjiencoding{#1}\else\romanencoding{#1}\fi}

```

\@@kenc@update \kanjiencoding コマンドのコードからもわかるように、\ck@encoding と \k@encoding が異なる場合、\kenc@update コマンドは \@@kenc@update コマンドと等しくなります。

\@@kenc@update コマンドは、そのエンコードでのデフォルト値を設定するためのコマンドです。欧文用の \@@enc@update コマンドでは、1221 行目と 1222 行目のような代入もしていますが、和文用にはコメントにしてあります。これらは \DeclareTextCommand や \ProvideTextCommand などエンコードごとに設定されるコマンドを使うための仕組みです。しかし、和文エンコードに依存するようなコマンドやマクロを作成することは、現時点では、ないと思います。

```

1220 \def\@@kenc@update{%
1221 %   \expandafter\let\csname\ck@encoding -cmd\endcsname\@changed@kcmd
1222 %   \expandafter\let\csname\k@encoding-cmd\endcsname\@current@cmd
1223   \default@KT
1224   \csname T@\k@encoding\endcsname
1225   \csname D@\k@encoding\endcsname
1226   \let\kenc@update\relax
1227   \let\ck@encoding\k@encoding
1228   \edef\tmp@item{{\k@encoding}}%
1229   \expandafter\expandafter\expandafter
1230   \inlist@\expandafter\tmp@item\expandafter{\kyenc@list}%
1231   \ifin@ \let\cy@encoding\k@encoding
1232   \else
1233     \expandafter\expandafter\expandafter
1234     \inlist@\expandafter\tmp@item\expandafter{\ktenc@list}%
1235     \ifin@ \let\ct@encoding\k@encoding
1236   \else

```

```

1237 \latexerror{KANJI Encoding scheme ‘\k@encoding’ unknown}\@eha
1238 \fi
1239 \fi
1240 }
1241 \let\kenc@update\relax

```

\@changed@kcmd \@changed@cmd の和文エンコーディングバージョン。

```

1242 \def\@changed@kcmd#1#2{%
1243   \ifx\protect\@typeset@protect
1244     \inmathwarn#1%
1245     \expandafter\ifx\csname\ck@encoding\string#1\endcsname\relax
1246       \expandafter\ifx\csname ?\string#1\endcsname\relax
1247         \expandafter\def\csname ?\string#1\endcsname{%
1248           \TextSymbolUnavailable#1%
1249         }%
1250       \fi
1251       \global\expandafter\let
1252         \csname\cf@encoding\string#1\expandafter\endcsname
1253         \csname ?\string#1\endcsname
1254       \fi
1255       \csname\ck@encoding\string#1%
1256       \expandafter\endcsname
1257     \else
1258       \noexpand#1%
1259     \fi}

```

7.2.7 ファミリの指定

\@notkfam \fontfamily コマンド内で使用するフラグです。@notkfam フラグは和文ファミリー
 \@notffam でなかったことを、@notffam フラグは欧文ファミリーでなかったことを示します。

```

1260 \newif\if@notkfam
1261 \newif\if@notffam

1262 \newif\if@tempwz

```

\romanfamily 書体のファミリーを指定するコマンドです。

\kanjifamily \kanjifamily と \romanfamily は与えられた引数が、和文あるいは欧文のファミリーとして正しいかのチェックは行なっていません。そのため、高速に動作をしますが、\kanjifamily に欧文ファミリーを指定したり、逆に \romanfamily に和文ファミリーを指定した場合は、エラーとなり、代用フォントかエラーフォントが使われます。

```

1263 \DeclareRobustCommand\romanfamily[1]{\edef\f@family{#1}}
1264 \DeclareRobustCommand\kanjifamily[1]{\edef\k@family{#1}}

```

\fontfamily は、指定された値によって、和文ファミリーか欧文ファミリー、あるいは両方のファミリーを切り替えます。和欧文ともに無効なファミリー名が指定された場合は、和欧文ともに代替書体が使用されます。

引数が \rmfamily のような名前与えられる可能性があるため、まず、これを展開したものを作ります。

また、和文ファミリと欧文ファミリのそれぞれになかったことを示すフラグを偽にセットします。

```
1265 \DeclareRobustCommand\fontfamily[1]{%
1266   \edef\tmp@item{#{1}}%
1267   \@notkfamfalse
1268   \@notffamfalse
```

次に、この引数が `\kfam@list` に登録されているかどうかを調べます。登録されていれば、`\k@family` にその値を入れます。

```
1269   \expandafter\expandafter\expandafter
1270   \inlist@\expandafter\tmp@item\expandafter{\kfam@list}%
1271   \ifin@ \edef\k@family{#1}%
```

そうでないときは、`\notkfam@list` に登録されているかどうかを調べます。登録されていれば、この引数は和文ファミリではありませんので、`\@notkfam` フラグを真にして、欧文ファミリのルーチンに移ります。

このとき、`\ffam@list` を調べるのではないことに注意してください。`\ffam@list` を調べ、これにないファミリを和文ファミリであるとする、たとえば、欧文ナールファミリが定義されているけれども、和文ナールファミリが未定義の場合、`\fontfamily{nar}` という指定は、`nar` が `\ffam@list` にだけ、登録されているため、和文書体をナールにすることができません。

逆に、`\kfam@list` に登録されていないからといって、`\k@family` に `nar` を設定すると、`cmr` のようなファミリも `\k@family` に設定される可能性があります。したがって、「欧文でない」を明示的に示す `\notkfam@list` を見る必要があります。

```
1272   \else
1273     \expandafter\expandafter\expandafter
1274     \inlist@\expandafter\tmp@item\expandafter{\notkfam@list}%
1275     \ifin@ \@notkfamtrue
```

`\notkfam@list` に登録されていない場合は、フォント定義ファイルが存在するかどうかを調べます。ファイルが存在する場合は、`\k@family` を変更します。ファイルが存在しない場合は、`\notkfam@list` に登録します。

`\kenc@list` に登録されているエンコードと、指定された和文ファミリの組合せのフォント定義ファイルが存在する場合は、`\k@family` に指定された値を入れます。

```
1276   \else
1277     \@tempzwfalse
1278     \def\fam@elt{\noexpand\fam@elt}%
1279     \message{(I search kanjifont definition file:)%
1280     \def\enc@elt<#1>{\message{.}%
1281       \edef\reserved@a{\lowercase{\noexpand\IfFileExists{##1#1.fd}}}%
1282       \reserved@a{\@tempzwtrue}{\relax}%
1283     \kenc@list
1284     \message{)}%
1285     \if@tempzw
1286       \edef\k@family{#1}%
```

つぎの部分が実行されるのは、和文ファミリとして認識できなかった場合です。この場合は、`\@notkfam` フラグを真にして、`\notkfam@list` に登録します。

```

1287     \else
1288     \notkfamtrue
1289     \xdef\notkfam@list{\notkfam@list\fam@elt<#1>}%
1290     \fi

    \kfam@list と \notkfam@list に登録されているかどうかを調べた \ifin@を閉じ
    ます。
1291     \fi\fi

    欧文ファミリの場合も、和文ファミリと同様の方法で確認をします。
1292     \expandafter\expandafter\expandafter
1293     \inlist@\expandafter\tmp@item\expandafter{\ffam@list}%
1294     \ifin@ \edef\f@family{#1}\else
1295     \expandafter\expandafter\expandafter
1296     \inlist@\expandafter\tmp@item\expandafter{\notffam@list}%
1297     \ifin@ \notffamtrue \else
1298     \@tempzwzfalse
1299     \def\fam@elt{\noexpand\fam@elt}%
1300     \message{(I search font definition file:}%
1301     \def\enc@elt<#1>{\message{.}%
1302     \edef\reserved@a{\lowercase{\noexpand\IfFileExists{##1#1.fd}}}%
1303     \reserved@a{\@tempzwztrue}{ }\relax}%
1304     \fenc@list
1305     \message{)}%
1306     \if@tempzwz
1307     \edef\f@family{#1}%
1308     \else
1309     \notffamtrue
1310     \xdef\notffam@list{\notffam@list\fam@elt<#1>}%
1311     \fi
1312     \fi\fi

    最後に、指定された文字列が、和文ファミリと欧文ファミリのいずれか、あるいは
    両方として認識されたかどうかを確認します。

    どちらも認識されていない場合は、ファミリの指定ミスですので、代用フォ
    ントを使うために、故意に指定された文字列をファミリに入れます。
1313     \if@notkfam\if@notffam
1314     \edef\k@family{#1}\edef\f@family{#1}%
1315     \fi\fi}
1316 \</plcore>

```

7.2.8 シリーズの指定（新 NFSS 対応）

\pltx@latex@level コミュニティ版 pL^AT_EX 2_ε 2020-02-02 での変更：ここから L^AT_EX 2_ε 2020-02-02 で拡張された新しい NFSS への対応コードが始まります。pL^AT_EX 2_ε のコードを本家 L^AT_EX 2_ε の機能に応じて切り替えます。

L^AT_EX 2_ε 2020-02-02 のうち、patch level 2 には latex3/latex2e#277 のバグがあり、patch level 4 には latex3/latex2e#293 のバグがありました。さらに開発版 L^AT_EX 2_ε では latex3/latex2e#291 の対策も施されています。

```

1317 \<*/plcore | platexrelease>

```

```

1318 \ifx\fontseriesforce\@undefined      % old
1319     \def\pltx@latex@level{0}
1320 \else                                  % 2020-02-02
1321     \ifx\@forced@seriestrue\@undefined
1322         \ifnum\patch@level<1\relax      % patch level 0
1323             \def\pltx@latex@level{1}% use \@reserveda
1324         \else                            % patch level 1, 2
1325             \def\pltx@latex@level{2}
1326         \fi
1327     \else
1328         \ifx\series@maybe@drop@one@m\@undefined % patch level 3, 4
1329             \def\pltx@latex@level{3}
1330         \else
1331             \ifx\series@maybe@drop@one@m@x\@undefined % patch level 5
1332                 \def\pltx@latex@level{4}
1333                 % anticipating LaTeX2e 'develop' branch (after 23b7244)
1334                 % this temporary code will be removed in the future
1335                 %\let\series@maybe@drop@one@m@x\series@maybe@drop@one@m
1336                 %\def\series@maybe@drop@one@m#1{%
1337                 %    \expandafter\series@maybe@drop@one@m@x\expandafter{#1}}
1338             \else
1339                 \def\pltx@latex@level{5}
1340             \fi
1341         \fi
1342     \fi
1343 \fi

```

ここでは、最低限どのバージョンの $\text{\LaTeX} 2_{\epsilon}$ 上でもフォーマット生成が成功するように `\catcode` トリックを使います。現在の主要なコードは

- $\text{\LaTeX} 2_{\epsilon}$ 2019-10-01 patch level 3 以前（従来の NFSS2）
- $\text{\LaTeX} 2_{\epsilon}$ の開発版（最新の develop ブランチ）

向けに最適化しており、他のバージョンへの対処は後回しにします。

```

1344 \edef\pltx@reset@catcode@trick{\catcode'\noexpand\~= \the\catcode'\~\relax}
1345 \def\pltx@temp@catcode@ix{\catcode'\~=9\relax}
1346 \def\pltx@temp@catcode@xiv{\catcode'\~=14\relax}
1347 \ifnum\pltx@latex@level<3\relax
1348     \pltx@temp@catcode@xiv % hide if-tokens
1349 \else
1350     \pltx@temp@catcode@ix % reveal if-tokens
1351 \fi
1352 </plcore | platexrelease>

```

`\delayed@k@adjustment` $\text{\LaTeX} 2_{\epsilon}$ 2021-06-01 で追加された `\delayed@f@adjustment` の和文版です。

```

1353 < *plcore | platexrelease>
1354 \ifx\delayed@f@adjustment\@undefined % old
1355     \let\delayed@k@adjustment\@undefined
1356 \else                                  % 2021-06-01
1357     \let\delayed@k@adjustment\@empty
1358 \fi
1359 </plcore | platexrelease>

```

`\if@forced@series@kanji` L^AT_EX 2_ε 2020-02-02 PL3 で追加された `\if@forced@series` フラグの和文版です。

```
1360 (*plcore|latexrelease)
1361 \ifx\@forced@series@true\@undefined % old
1362 \let\@forced@series@kanjifalse\@undefined
1363 \let\@forced@series@kanjitrue\@undefined
1364 \else % 2020-02-02 PL3 or later
1365 \expandafter\newif\csname if@forced@series@kanji\endcsname
1366 \fi
1367 </plcore|latexrelease>
```

`\romanseries` 書体のシリーズを指定するコマンドです。`\fontseries` コマンドは和欧文の両方に影響します。

`\fontseries` 2019 年までは無条件に指定されたとおりのシリーズを選択していましたが、L^AT_EX 2_ε 2020-02-02 以降では、`\DeclareFontSeriesChangeRule` によって宣言された「シリーズ更新規則」に基づきシリーズを選択します。

L^AT_EX 2_ε 2021-06-01 以降では、シリーズの更新を `\selectfont` まで遅らせます。

```
1368 (*plcore|latexrelease)
1369 \ifx\fontseriesforce\@undefined % old
1370 \DeclareRobustCommand\romanseries[1]{\edef\f@series{#1}}
1371 \DeclareRobustCommand\kanjiseriess[1]{\edef\k@series{#1}}
1372 \DeclareRobustCommand\fontseries[1]{\kanjiseriess{#1}\romanseriess{#1}}
1373 \else % 2020-02-02
1374 \ifx\delayed@f@adjustment\@undefined % --- for <= 2020-10-01 BEGIN
1375 \DeclareRobustCommand\romanseriess[1]{\@forced@seriesfalse\merge@font@series{#1}}
1376 \DeclareRobustCommand\kanjiseriess[1]{\@forced@series@kanjifalse\merge@kanji@series{#1}}
1377 \DeclareRobustCommand\fontseries[1]{\kanjiseriess{#1}\romanseriess{#1}}
1378 \else % --- for <= 2020-10-01 END & for >= 2021-06-01 BEGIN
1379 \DeclareRobustCommand\romanseriess[1]{\@forced@seriesfalse
1380 \expandafter\def\expandafter\delayed@f@adjustment\expandafter
1381 {\delayed@f@adjustment\delayed@merge@font@series{#1}}}
1382 \DeclareRobustCommand\kanjiseriess[1]{\@forced@series@kanjifalse
1383 \expandafter\def\expandafter\delayed@k@adjustment\expandafter
1384 {\delayed@k@adjustment\delayed@merge@kanji@series{#1}}}
1385 \DeclareRobustCommand\fontseries[1]{\kanjiseriess{#1}\romanseriess{#1}}
1386 \fi % --- for >= 2021-06-01 END
1387 \fi
```

`\romanseriessforce` 無条件にシリーズを変更します。

`\kanjiseriessforce` 1388 `\ifx\fontseriesforce\@undefined` % old

```
\fontseriesforce
1389 \let\romanseriessforce\@undefined
1390 \let\kanjiseriessforce\@undefined
1391 \else % 2020-02-02
1392 \ifx\delayed@f@adjustment\@undefined % --- for <= 2020-10-01 BEGIN
1393 \DeclareRobustCommand\romanseriessforce[1]{\@forced@series@true\edef\f@series{#1}}
1394 \DeclareRobustCommand\kanjiseriessforce[1]{\@forced@series@kanjitrue\edef\k@series{#1}}
1395 \DeclareRobustCommand\fontseriesforce[1]{\kanjiseriessforce{#1}\romanseriessforce{#1}}
1396 \else % --- for <= 2020-10-01 END & for >= 2021-06-01 BEGIN
1397 \DeclareRobustCommand\romanseriessforce[1]{\@forced@series@true
1398 \expandafter\def\expandafter\delayed@f@adjustment\expandafter
1399 {\delayed@f@adjustment\edef\f@series{#1}}}
```

```

1400 \DeclareRobustCommand\kanjiseriessforce[1]{\@forced@series@kanjittrue
1401   \expandafter\def\expandafter\delayed@k@adjustment\expandafter
1402     {\delayed@k@adjustment\edef\k@series{#1}}}
1403 \DeclareRobustCommand\fontseriesforce[1]{\kanjiseriessforce{#1}\romanseriessforce{#1}}
1404 \fi          % --- for >= 2021-06-01 END
1405 \fi

```

\merge@kanji@series \merge@font@series の和文版です。

```

\merge@kanji@series@406 \ifx\fontseriesforce\@undefined % old
1407 \let\merge@kanji@series\@undefined
\set@target@series@kanji
1408 \let\merge@kanji@series\@undefined
1409 \let\set@target@series@kanji\@undefined
1410 \else          % 2020-02-02
1411 \def\merge@kanji@series#1{%
1412   \expandafter\expandafter\expandafter
1413     \merge@kanji@series@
1414       \csname series@\k@series @#1\endcsname
1415       {#1}%
1416       \@nil
1417 }
1418 \def\merge@kanji@series@#1#2#3\@nil{%
1419   \def\reserved@a{#3}%
1420   \ifx\reserved@a\@empty

```

シリーズ更新規則がない場合：#2 が要求シリーズであり、これを使う。

```

1421   \set@target@series@kanji{#2}%
1422   \else
1423   %^^A [TODO] BEGIN
1424   %^^A   LaTeX2e 2021-06-01 では |\maybe@load@fontshape| は削除される。
1425   %^^A   理由：処理が |\selectfont| まで遅れるので不要とのこと。
1426   %^^A   しかし、なぜか ltfsaxes.dtx で rollback の対処が無いような？
1427   \begingroup\let\f@encoding\k@encoding\let\f@family\k@family
1428     \maybe@load@fontshape\endgroup
1429   %^^A [TODO] END
1430   \edef\reserved@a{\k@encoding /\k@family /#1/\k@shape}%
1431   \ifcsname \reserved@a \endcsname

```

シリーズ更新規則に基づく新シリーズ #1 が利用可能：

```

1432   \set@target@series@kanji{#1}%
1433   \else
1434   \ifcsname \k@encoding /\k@family /#2/\k@shape \endcsname

```

シリーズ更新規則に基づく代替シリーズ #2 が利用可能：

```

1435   \set@target@series@kanji{#2}%
1436   {\let\curr@fontshape\curr@kfontshape\@font@shape@subst@warning}%
1437   \else

```

いずれも利用不可：要求シリーズ #3 を使う。

```

1438   \set@target@series@kanji{#3}%
1439   {\let\curr@fontshape\curr@kfontshape\@font@shape@subst@warning}%
1440   \fi
1441   \fi
1442   \fi
1443 }

```

```

1444 \def\set@target@series@kanji#1{%
1445   \edef\k@series{#1}%
1446   \series@maybe@drop@one@m\k@series\k@series
1447 }
1448 \fi

ji@series@without@substitution \merge@font@series@without@substitution の和文版です。
i@series@without@substitution@449 \ifx\merge@font@series@without@substitution\undefined % old
\delayed@merge@kanji@series 1450 \let\merge@kanji@series@without@substitution\undefined
1451 \let\merge@kanji@series@without@substitution@\undefined
1452 \let\delayed@merge@kanji@series\undefined
1453 \else % 2021-06-01
1454 \def\merge@kanji@series@without@substitution#1{%
1455   \expandafter\expandafter\expandafter
1456   \merge@kanji@series@without@substitution@
1457   \csname series@\k@series @#1\endcsname
1458   {#1}%
1459   \@nil
1460 }
1461 \def\merge@kanji@series@without@substitution#1#2#3\@nil{%
1462   \def\reserved@a{#3}%
1463   \ifx\reserved@a\empty
1464     \set@target@series@kanji{#2}%
1465   \else
1466     \set@target@series@kanji{#1}%
1467   \fi
1468 }
1469 \let\delayed@merge@kanji@series\merge@kanji@series@without@substitution
1470 \fi
1471 </plcore | platexrelease>

```

7.2.9 シェイプの指定（新 NFSS 対応）

コミュニティ版 p \LaTeX 2 ϵ 2020-04-12 での変更：従来は、`\itshape` などの命令を実行すると

```

LaTeX Font Warning: Font shape 'JT1/mc/m/it' undefined
(Font)                using 'JT1/mc/m/n' instead on input line 4.
LaTeX Font Warning: Font shape 'JY1/mc/m/it' undefined
(Font)                using 'JY1/mc/m/n' instead on input line 4.

```

のような警告を発していました。これは以下の理由によります。

- \LaTeX 2 ϵ が定義する `\itshape` などのシェイプ変更命令は内部で `\fontshape` を呼び出す。
- p \LaTeX 2 ϵ では、`\fontshape` を欧文書体だけでなく和文書体も変更するように再定義する。
- しかし、和文書体のシェイプはほとんど“n”しか用いられず、`\DeclareFontShape` での定義も“n”しか与えられないことが多い。

- 結果的に、欧文書体のシェイプを変更するつもりでも「和文書体のシェイプが未定義」という警告が出てしまう。

そこで、和文書体のシェイプが未定義の場合は `\fontshape` 及び `\fontshapeforce` が和文書体には影響せず、欧文書体のシェイプのみを変更するように改良します。

`\if@shape@roman@kanji` と欧文の両方に影響しようとする `\fontshape` コマンド実行中に真になるフラグです。`\fontshapeforce` は実装が単純なので、このフラグは使っていません。

```
1472 < *plcore | latexrelease >
1473 \ifx\@shape@roman@kanjittrue\@undefined % just in case
1474   \expandafter\newif\csname if@shape@roman@kanji\endcsname
1475 \fi
1476 < /plcore | latexrelease >
```

`\romanshape` 書体のシェイプを指定するコマンドです。`\fontshape` コマンドは和欧文の両方に `\kanjishape` 影響します。

`\fontshape` 2019 年までは無条件に指定されたとおりのシェイプを選択していましたが、 $\text{\LaTeX 2}_{\epsilon}$ 2020-02-02 以降では、`\DeclareFontShapeChangeRule` によって宣言された「シェイプ更新規則」に基づきシェイプを選択します。

$\text{\LaTeX 2}_{\epsilon}$ 2021-06-01 以降では、シェイプの更新を `\selectfont` まで遅らせます。

```
1477 < latexrelease > \plIncludeInRelease{2020/04/12}{\fontshape}
1478 < latexrelease > {No \k@shape update if unavailable}%
1479 < *plcore | latexrelease >
1480 \ifx\fontshapeforce\@undefined % old
1481 \DeclareRobustCommand\romanshape[1]{\edef\f@shape{#1}}
1482 \DeclareRobustCommand\kanjishape[1]{\edef\k@shape{#1}}
1483 \DeclareRobustCommand\fontshape[1]{%
1484   \set@safe@kanji@shape{#1}%
1485   \edef\f@shape{#1}%
1486 }
1487 \else % 2020-02-02
1488 \ifx\delayed@f@adjustment\@undefined % --- for <= 2020-10-01 BEGIN
1489 \DeclareRobustCommand\romanshape[1]{\merge@font@shape{#1}}
1490 \DeclareRobustCommand\kanjishape[1]{\merge@kanji@shape{#1}}
1491 \DeclareRobustCommand\fontshape[1]{%
1492   \@shape@roman@kanjittrue
1493   \kanjishape{#1}\romanshape{#1}%
1494   \@shape@roman@kanjifalse}
1495 \else % --- for <= 2020-10-01 END & for >= 2021-06-01 BEGIN
1496 \DeclareRobustCommand\romanshape[1]{%
1497   \expandafter\def\expandafter\delayed@f@adjustment\expandafter
1498     {\delayed@f@adjustment\delayed@merge@font@shape{#1}}
1499 \DeclareRobustCommand\kanjishape[1]{%
1500   \expandafter\def\expandafter\delayed@k@adjustment\expandafter
1501     {\delayed@k@adjustment\delayed@merge@kanji@shape{#1}}
1502 \DeclareRobustCommand\fontshape[1]{%
1503   \romanshape{#1}%
1504   \expandafter\def\expandafter\delayed@k@adjustment\expandafter
1505     {\delayed@k@adjustment\@shape@roman@kanjittrue
```

```

1506      \delayed@merge@kanji@shape{#1}\@shape@roman@kanjifalse}}
1507 \fi      % --- for >= 2021-06-01 END
1508 \fi
1509 </plcore | platexrelease>
1510 <platexrelease>\plEndIncludeInRelease
1511 <platexrelease>\plIncludeInRelease{0000/00/00}{\fontshape}
1512 <platexrelease>      {ASCII Corporation / TeXJP original}%
1513 <platexrelease>\ifx\fontshapeforce\@undefined % old
1514 <platexrelease>\DeclareRobustCommand\romanshape[1]{\edef\f@shape{#1}}
1515 <platexrelease>\DeclareRobustCommand\kanjishape[1]{\edef\k@shape{#1}}
1516 <platexrelease>\DeclareRobustCommand\fontshape[1]{\kanjishape{#1}\romanshape{#1}}
1517 <platexrelease>\else % 2020-02-02
1518 <platexrelease>\DeclareRobustCommand\romanshape[1]{\merge@font@shape{#1}}
1519 <platexrelease>\DeclareRobustCommand\kanjishape[1]{\merge@kanji@shape{#1}}
1520 <platexrelease>\DeclareRobustCommand\fontshape[1]{\kanjishape{#1}\romanshape{#1}}
1521 <platexrelease>\fi
1522 <platexrelease>\plEndIncludeInRelease

```

\romanshapeforce 無条件にシェイプを変更します。

```

\kanjishapeforce 523 <platexrelease>\plIncludeInRelease{2020/04/12}{\fontshapeforce}
\fontshapeforce 1524 <platexrelease>      {No \k@shape update if unavailable}%
1525 <*plcore | platexrelease>
1526 \ifx\fontshapeforce\@undefined % old
1527 \let\romanshapeforce\@undefined
1528 \let\kanjishapeforce\@undefined
1529 \else % 2020-02-02
1530 \ifx\delayed@f@adjustment\@undefined % --- for <= 2020-10-01 BEGIN
1531 \DeclareRobustCommand\romanshapeforce[1]{\edef\f@shape{#1}}
1532 \DeclareRobustCommand\kanjishapeforce[1]{\edef\k@shape{#1}}
1533 \DeclareRobustCommand\fontshapeforce[1]{%
1534   \set@safe@kanji@shape{#1}{}%
1535   \edef\f@shape{#1}%
1536 }
1537 \else % --- for <= 2020-10-01 END & for >= 2021-06-01 BEGIN
1538 \DeclareRobustCommand\romanshapeforce[1]{%
1539   \expandafter\def\expandafter\delayed@f@adjustment\expandafter
1540     {\delayed@f@adjustment\edef\f@shape{#1}}
1541 \DeclareRobustCommand\kanjishapeforce[1]{%
1542   \expandafter\def\expandafter\delayed@k@adjustment\expandafter
1543     {\delayed@k@adjustment\edef\k@shape{#1}}
1544 \DeclareRobustCommand\fontshapeforce[1]{%
1545   \expandafter\def\expandafter\delayed@k@adjustment\expandafter
1546     {\delayed@k@adjustment\set@safe@kanji@shape{#1}{}%
1547     \expandafter\def\expandafter\delayed@f@adjustment\expandafter
1548       {\delayed@f@adjustment\edef\f@shape{#1}}%
1549 }
1550 \fi % --- for >= 2021-06-01 END
1551 \fi
1552 </plcore | platexrelease>
1553 <platexrelease>\plEndIncludeInRelease
1554 <platexrelease>\plIncludeInRelease{0000/00/00}{\fontshapeforce}
1555 <platexrelease>      {ASCII Corporation / TeXJP original}%
1556 <platexrelease>\ifx\fontshapeforce\@undefined % old
1557 <platexrelease>\let\romanshapeforce\@undefined

```



```

1558 <platexrelease>\let\kanjishapeforce\@undefined
1559 <platexrelease>\else % 2020-02-02
1560 <platexrelease>\DeclareRobustCommand\romanshapeforce[1]{\edef\f@shape{#1}}
1561 <platexrelease>\DeclareRobustCommand\kanjishapeforce[1]{\edef\k@shape{#1}}
1562 <platexrelease>\DeclareRobustCommand\fontshapeforce[1]{\kanjishapeforce{#1}\romanshapeforce{#1}}
1563 <platexrelease>\fi
1564 <platexrelease>\plEndIncludeInRelease

```

\merge@kanji@shape \merge@font@shape の和文版です。

```

\merge@kanji@shape@1565 <platexrelease>\plIncludeInRelease{2020/04/12}{\merge@kanji@shape@
1566 <platexrelease> {No \k@shape update if unavailable}%
1567 *plcore | platexrelease)
1568 \ifx\fontseriesforce\@undefined % old
1569 \let\merge@kanji@shape\@undefined
1570 \let\merge@kanji@shape@\@undefined
1571 \else % 2020-02-02
1572 \def\merge@kanji@shape#1{%
1573 \expandafter\expandafter\expandafter
1574 \merge@kanji@shape@
1575 \csname shape@k@shape @#1\endcsname
1576 {#1}%
1577 \@nil
1578 }
1579 \def\merge@kanji@shape@#1#2#3\@nil{%
1580 \def\reserved@a{#3}%
1581 \ifx\reserved@a\@empty

```

シェイプ更新規則がない場合：#2 が要求シェイプである。

\fontshape の下請けなら、#2 が利用可能かどうか予めチェックする。

\kanjishape の下請けなら、#2 を使う。

```

1582 \if@shape@roman@kanji
1583 \set@safe@kanji@shape{#2}{}%
1584 \else
1585 \edef\k@shape{#2}%
1586 \fi
1587 \else
1588 %^^A [TODO] BEGIN
1589 %^^A LaTeX2e 2021-06-01 では |\maybe@load@fontshape| は削除される。
1590 %^^A 理由：処理が |\selectfont| まで遅れるので不要とのこと。
1591 %^^A しかし、なぜか ltfssaxes.dtx で rollback の対処が無いような？
1592 \begingroup\let\f@encoding\k@encoding\let\f@family\k@family
1593 \maybe@load@fontshape\endgroup
1594 %^^A [TODO] END
1595 \edef\reserved@a{\k@encoding /\k@family /\k@series/#1}%
1596 \ifcsname \reserved@a\endcsname

```

シェイプ更新規則に基づく新シェイプ #1 が利用可能：

```

1597 \edef\k@shape{#1}%
1598 \else
1599 \ifcsname \k@encoding /\k@family /\k@series/#2\endcsname

```

シェイプ更新規則に基づく代替シェイプ #2 が利用可能：

```

1600 \edef\k@shape{#2}%

```

```

1601         {\let\curr@fontshape\curr@kfontshape\@font@shape@subst@warning}%
1602         \else

いずれも利用不可：要求シェイプ #3 について
\fontshape の下請けなら、#3 が利用可能かどうか予めチェックする。
\kanjishape の下請けなら、#3 を使う。

1603         \if@shape@roman@kanji
1604         \set@safe@kanji@shape{#3}%
1605         {\let\curr@fontshape\curr@kfontshape\@font@shape@subst@warning}%
1606         \else
1607         \edef\k@shape{#3}%
1608         {\let\curr@fontshape\curr@kfontshape\@font@shape@subst@warning}%
1609         \fi
1610         \fi
1611         \fi
1612         \fi
1613     }
1614 \fi
1615 </plcore | platexrelease>
1616 <platexrelease>\plEndIncludeInRelease
1617 <platexrelease>\plIncludeInRelease{0000/00/00}{\merge@kanji@shape@
1618 <platexrelease>                {ASCII Corporation / TeXJP original}%
1619 <platexrelease>\ifx\fontseriesforce\@undefined % old
1620 <platexrelease>\let\merge@kanji@shape\@undefined
1621 <platexrelease>\let\merge@kanji@shape\@undefined
1622 <platexrelease>\else % 2020-02-02
1623 <platexrelease>\def\merge@kanji@shape#1{%
1624 <platexrelease> \expandafter\expandafter\expandafter
1625 <platexrelease> \merge@kanji@shape@
1626 <platexrelease> \csname shape@k@shape @#1\endcsname
1627 <platexrelease> {#1}%
1628 <platexrelease> \@nil
1629 <platexrelease>}
1630 <platexrelease>\def\merge@kanji@shape@#1#2#3\@nil{%
1631 <platexrelease> \def\reserved@a{#3}%
1632 <platexrelease> \ifx\reserved@a\@empty
1633 <platexrelease> \edef\k@shape{#2}%
1634 <platexrelease> \else
1635 <platexrelease> \begingroup\let\f@encoding\k@encoding\let\f@family\k@family
1636 <platexrelease> \maybe@load@fontshape@endgroup
1637 <platexrelease> \edef\reserved@a{\k@encoding /\k@family /\k@series/#1}%
1638 <platexrelease> \ifcsname \reserved@a\endcsname
1639 <platexrelease> \edef\k@shape{#1}%
1640 <platexrelease> \else
1641 <platexrelease> \ifcsname \k@encoding /\k@family /\k@series/#2\endcsname
1642 <platexrelease> \edef\k@shape{#2}%
1643 <platexrelease> {\let\curr@fontshape\curr@kfontshape\@font@shape@subst@warning}%
1644 <platexrelease> \else
1645 <platexrelease> \edef\k@shape{#3}%
1646 <platexrelease> {\let\curr@fontshape\curr@kfontshape\@font@shape@subst@warning}%
1647 <platexrelease> \fi
1648 <platexrelease> \fi
1649 <platexrelease> \fi
1650 <platexrelease>}

```

```

1651 <latexrelease>\fi
1652 <latexrelease>\plEndIncludeInRelease

\kanji@shape@without@substitution \merge@font@shape@without@substitution の和文版です。
\kanji@shape@without@substitution@653 <*plcore | latexrelease>
\delayed@merge@kanji@shape1654 \ifx\merge@font@shape@without@substitution\@undefined % old
1655 \let\merge@kanji@shape@without@substitution\@undefined
1656 \let\merge@kanji@shape@without@substitution\@undefined
1657 \let\delayed@merge@kanji@shape\@undefined
1658 \else % 2021-06-01
1659 \def\merge@kanji@shape@without@substitution#1{%
1660 \expandafter\expandafter\expandafter
1661 \merge@kanji@shape@without@substitution@
1662 \csname shape@k@shape @#1\endcsname
1663 {#1}%
1664 \@nil
1665 }
1666 \def\merge@kanji@shape@without@substitution#1#2#3\@nil{%
1667 \def\reserved@a{#3}%
1668 \ifx\reserved@a\@empty
1669 \edef\k@shape{#2}%
1670 \else
1671 \edef\k@shape{#1}%
1672 \fi
1673 }
1674 \let\delayed@merge@kanji@shape\merge@kanji@shape@without@substitution
1675 \fi
1676 </plcore | latexrelease>

\set@safe@kanji@shape 和文シェープが利用可能かどうか予めチェックしてから設定します。
\kanji@shape@nochange@info1677 <latexrelease>\plIncludeInRelease{2020/04/12}{\set@safe@kanji@shape}
1678 <latexrelease> {No \k@shape update if unavailable}%
1679 <*plcore | latexrelease>
1680 \def\set@safe@kanji@shape#1#2{%
1681 \edef\reserved@b{\k@encoding /\k@family /\k@series/#1}%
1682 \ifcsname \reserved@b\endcsname
1683 \edef\k@shape{#1}%
1684 #2%
1685 \else
1686 \@kanji@shape@nochange@info{\reserved@b}%
1687 \fi
1688 }
1689 \def\@kanji@shape@nochange@info#1{%
1690 \@font@info{Kanji font shape ‘#1’ undefined\MessageBreak
1691 No change}%
1692 }
1693 </plcore | latexrelease>
1694 <latexrelease>\plEndIncludeInRelease
1695 <latexrelease>\plIncludeInRelease{0000/00/00}{\set@safe@kanji@shape}
1696 <latexrelease> {ASCII Corporation original}%
1697 <latexrelease>\let\set@safe@kanji@shape\@undefined
1698 <latexrelease>\let\@kanji@shape@nochange@info\@undefined
1699 <latexrelease>\plEndIncludeInRelease

```

7.2.10 書体の切り替え（新 NFSS 対応）

`\usekanji` 書体属性を一度に指定するコマンドです。和文書体には `\usekanji` を、欧文書体には `\useroman` は `\useroman` を指定してください。

`\usefont` `\usefont` コマンドは、第一引数で指定されるエンコードによって、和文または欧文フォントを切り替えます。

```
1700 <latexrelease>\plIncludeInRelease{2020/02/02}{\usefont}
1701 <latexrelease> {Don't call \fontseries or \fontshape}%
1702 (*plcore| latexrelease)
1703 \ifx\set@target@series\@undefined % old
1704 \DeclareRobustCommand\usekanji[4]{\kanjiencoding{#1}%
1705   \edef\k@family{#2}%
1706   \edef\k@series{#3}%
1707   \edef\k@shape{#4}\selectfont
1708   \ignorespaces}
1709 \DeclareRobustCommand\useroman[4]{\romanencoding{#1}%
1710   \edef\f@family{#2}%
1711   \edef\f@series{#3}%
1712   \edef\f@shape{#4}\selectfont
1713   \ignorespaces}
1714 \else % 2020-02-02
1715 \ifx\delayed@f@adjustment\@undefined % --- for <= 2020-10-01 BEGIN
1716 \DeclareRobustCommand\usekanji[4]{\kanjiencoding{#1}%
1717   \edef\k@family{#2}%
1718   \set@target@series@kanji{#3}%
1719   \edef\k@shape{#4}\selectfont
1720   \ignorespaces}
1721 \DeclareRobustCommand\useroman[4]{\romanencoding{#1}%
1722   \edef\f@family{#2}%
1723   \set@target@series{#3}%
1724   \edef\f@shape{#4}\selectfont
1725   \ignorespaces}
1726 \else % --- for <= 2020-10-01 END & for >= 2021-06-01 BEGIN
1727 \DeclareRobustCommand\usekanji[4]{\kanjiencoding{#1}%
1728   \edef\k@family{#2}%
1729   \set@target@series@kanji{#3}%
1730   \edef\k@shape{#4}%
1731   \let\delayed@k@adjustment\@empty
1732   \selectfont
1733   \ignorespaces}
1734 \DeclareRobustCommand\useroman[4]{\romanencoding{#1}%
1735   \edef\f@family{#2}%
1736   \set@target@series{#3}%
1737   \edef\f@shape{#4}%
1738   \let\delayed@f@adjustment\@empty
1739   \selectfont
1740   \ignorespaces}
1741 \fi % --- for >= 2021-06-01 END
1742 \fi % done
1743 \DeclareRobustCommand\usefont[4]{%
1744   \edef\tmp@item{#1}%
1745   \expandafter\expandafter\expandafter
1746   \inlist@\expandafter\tmp@item\expandafter{\kenc@list}%

```

```

1747 \ifin@ \usekanji{#1}{#2}{#3}{#4}%
1748 \else\useroman{#1}{#2}{#3}{#4}%
1749 \fi}
1750 </plcore | latexrelease>
1751 <latexrelease>\plEndIncludeInRelease
1752 <latexrelease>\plIncludeInRelease{2019/10/01}{\usefont}
1753 <latexrelease> {Make robust}%
1754 <latexrelease>\DeclareRobustCommand\usekanji[4]{%
1755 <latexrelease> \kanjiencoding{#1}\kanjifamily{#2}\kanjiserie{#3}\kanjishape{#4}%
1756 <latexrelease> \selectfont\ignorespaces}
1757 <latexrelease>\DeclareRobustCommand\useroman[4]{%
1758 <latexrelease> \romanencoding{#1}\romanfamily{#2}\romanserie{#3}\romanshape{#4}%
1759 <latexrelease> \selectfont\ignorespaces}
1760 <latexrelease>\DeclareRobustCommand\usefont[4]{%
1761 <latexrelease> \edef\tmp@item{{#1}}%
1762 <latexrelease> \expandafter\expandafter\expandafter
1763 <latexrelease> \inlist@\expandafter\tmp@item\expandafter{\kenc@list}%
1764 <latexrelease> \ifin@ \usekanji{#1}{#2}{#3}{#4}%
1765 <latexrelease> \else\useroman{#1}{#2}{#3}{#4}%
1766 <latexrelease> \fi}
1767 <latexrelease>\plEndIncludeInRelease
1768 <latexrelease>\plIncludeInRelease{0000/00/00}{\usefont}
1769 <latexrelease> {ASCII Corporation original}%
1770 <latexrelease>\def\usekanji#1#2#3#4{%
1771 <latexrelease> \kanjiencoding{#1}\kanjifamily{#2}\kanjiserie{#3}\kanjishape{#4}%
1772 <latexrelease> \selectfont\ignorespaces}
1773 <latexrelease>\def\useroman#1#2#3#4{%
1774 <latexrelease> \romanencoding{#1}\romanfamily{#2}\romanserie{#3}\romanshape{#4}%
1775 <latexrelease> \selectfont\ignorespaces}
1776 <latexrelease>\def\usefont#1#2#3#4{%
1777 <latexrelease> \edef\tmp@item{{#1}}%
1778 <latexrelease> \expandafter\expandafter\expandafter
1779 <latexrelease> \inlist@\expandafter\tmp@item\expandafter{\kenc@list}%
1780 <latexrelease> \ifin@ \usekanji{#1}{#2}{#3}{#4}%
1781 <latexrelease> \else\useroman{#1}{#2}{#3}{#4}%
1782 <latexrelease> \fi}
1783 <latexrelease>\expandafter \let \csname usekanji \endcsname \@undefined
1784 <latexrelease>\expandafter \let \csname useroman \endcsname \@undefined
1785 <latexrelease>\expandafter \let \csname usefont \endcsname \@undefined
1786 <latexrelease>\plEndIncludeInRelease

```

`\normalfont` 書体をデフォルト値にするコマンドです。和文書体もデフォルト値になるように再定義しています。ただし高速化のため、`\usekanji` と `\useroman` を展開し、`\selectfont` を一度しか呼び出さないようにしています。

LaTeX 2_ε 2020-02-02 patch level 2 で新設されたフック `\@defaultfamilyhook` を使うことで、元の定義を上書きする必要がなくなりました。（注意：アスキー版の末尾にあった `\ignorespaces` を削除することで、元の LaTeX 2_ε と互換になりました。ltfssini.dtx 1995/10/16 v3.0f の変更も参考。）

LaTeX 2_ε 2020-10-01 では `\AddToHook` が使えます。

```

1787 <latexrelease>\plIncludeInRelease{2020/04/12}{\normalfont}
1788 <latexrelease> {Use \@defaultfamilyhook}%

```

```

1789 <|plcore| platexrelease>
1790 \ifnum\pltx@newhook@avail=\z@ % --- for <= 2020-02-02 BEGIN
1791 \ifx\@defaultfamilyhook\@undefined % old
1792 \DeclareRobustCommand\normalfont{%
1793   \kanjiencoding{\kanjiencodingdefault}%
1794   \edef\k@family{\kanjifamilydefault}%
1795   \edef\k@series{\kanjiseriessdefault}%
1796   \edef\k@shape{\kanjishapedefault}%
1797   \romanencoding{\encodingdefault}%
1798   \edef\f@family{\familydefault}%
1799   \edef\f@series{\seriesdefault}%
1800   \edef\f@shape{\shapedefault}%
1801   \selectfont}
1802 \else % 2020-02-02 PL2
1803 <|platexrelease>\DeclareRobustCommand\normalfont{%
1804 <|platexrelease>   \fontencoding\encodingdefault
1805 <|platexrelease>   \edef\f@family{\familydefault}%
1806 <|platexrelease>   \edef\f@series{\seriesdefault}%
1807 <|platexrelease>   \edef\f@shape{\shapedefault}%
1808 <|platexrelease>   \@defaultfamilyhook
1809 <|platexrelease>   \selectfont}
1810 \g@addto@macro\@defaultfamilyhook{%
1811   \kanjiencoding{\kanjiencodingdefault}%
1812   \edef\k@family{\kanjifamilydefault}%
1813   \edef\k@series{\kanjiseriessdefault}%
1814   \edef\k@shape{\kanjishapedefault}%
1815 }
1816 \fi % done
1817 \else % --- for <= 2020-02-02 END & for >= 2020-10-01 BEGIN
1818 \ifx\delayed@f@adjustment\@undefined % --- for == 2020-10-01 BEGIN
1819 <|platexrelease>\DeclareRobustCommand\normalfont{%
1820 <|platexrelease>   \fontencoding\encodingdefault
1821 <|platexrelease>   \edef\f@family{\familydefault}%
1822 <|platexrelease>   \edef\f@series{\seriesdefault}%
1823 <|platexrelease>   \edef\f@shape{\shapedefault}%
1824 <|platexrelease>   \UseHook{normalfont}%
1825 <|platexrelease>   \@defaultfamilyhook % hookname from 2020/02 will vanish
1826 <|platexrelease>   \selectfont}
1827 \AddToHook{normalfont}{%
1828   \kanjiencoding{\kanjiencodingdefault}%
1829   \edef\k@family{\kanjifamilydefault}%
1830   \edef\k@series{\kanjiseriessdefault}%
1831   \edef\k@shape{\kanjishapedefault}%
1832 }
1833 \else % --- for == 2020-10-01 END & for >= 2021-06-01 BEGIN
1834 <|platexrelease>\DeclareRobustCommand\normalfont{%
1835 <|platexrelease>   \fontencoding\encodingdefault
1836 <|platexrelease>   \edef\f@family{\familydefault}%
1837 <|platexrelease>   \edef\f@series{\seriesdefault}%
1838 <|platexrelease>   \edef\f@shape{\shapedefault}%
1839 <|platexrelease>   \let\delayed@f@adjustment\@empty
1840 <|platexrelease>   \UseHook{normalfont}%
1841 <|platexrelease>   \@defaultfamilyhook % hookname from 2020/02 will vanish
1842 <|platexrelease>   \selectfont}

```

```

1843 \AddToHook{normalfont}{%
1844     \kanjiencoding{\kanjiencodingdefault}%
1845     \edef\k@family{\kanjifamilydefault}%
1846     \edef\k@series{\kanjiseriessdefault}%
1847     \edef\k@shape{\kanjishapedefault}%
1848     \let\delayed@k@adjustment\@empty
1849 }
1850 \fi          % --- for >= 2021-06-01 END
1851 \fi          % --- for >= 2020-10-01 END
1852 \edef\@@@temp{\detokenize{nullfont}}\edef\@@@tempa{\fontname\jfont}
1853 \unless\ifx\@@@temp\@@@tempa\adjustbaseline\fi
1854 \let\@@@temp\undefined\let\@@@tempa\undefined
1855 \let\reset@font\normalfont
1856 </plcore| platexrelease>
1857 <platexrelease>\plEndIncludeInRelease
1858 <platexrelease>\plIncludeInRelease{2020/02/02}{\normalfont}
1859 <platexrelease>          {Don't call \fontseries or \fontshape}%
1860 <platexrelease>\DeclareRobustCommand\normalfont{%
1861 <platexrelease>    \kanjiencoding{\kanjiencodingdefault}%
1862 <platexrelease>    \edef\k@family{\kanjifamilydefault}%
1863 <platexrelease>    \edef\k@series{\kanjiseriessdefault}%
1864 <platexrelease>    \edef\k@shape{\kanjishapedefault}%
1865 <platexrelease>    \romanencoding{\encodingdefault}%
1866 <platexrelease>    \edef\f@family{\familydefault}%
1867 <platexrelease>    \edef\f@series{\seriesdefault}%
1868 <platexrelease>    \edef\f@shape{\shapedefault}%
1869 <platexrelease>    \selectfont\ignorespaces}
1870 <platexrelease>\adjustbaseline
1871 <platexrelease>\let\reset@font\normalfont
1872 <platexrelease>\plEndIncludeInRelease
1873 <platexrelease>\plIncludeInRelease{0000/00/00}{\normalfont}
1874 <platexrelease>          {ASCII Corporation original}%
1875 <platexrelease>\DeclareRobustCommand\normalfont{%
1876 <platexrelease>    \kanjiencoding{\kanjiencodingdefault}%
1877 <platexrelease>    \kanjifamily{\kanjifamilydefault}%
1878 <platexrelease>    \kanjiseriess{\kanjiseriessdefault}%
1879 <platexrelease>    \kanjishape{\kanjishapedefault}%
1880 <platexrelease>    \romanencoding{\encodingdefault}%
1881 <platexrelease>    \romanfamily{\familydefault}%
1882 <platexrelease>    \romanseries{\seriesdefault}%
1883 <platexrelease>    \romanshape{\shapedefault}%
1884 <platexrelease>    \selectfont\ignorespaces}
1885 <platexrelease>\adjustbaseline
1886 <platexrelease>\let\reset@font\normalfont
1887 <platexrelease>\plEndIncludeInRelease

```

\bfseries@mc L^AT_EX 2_ε 2020-02-02 では、欧文フォントについて「ファミリーごとの実際のシリーズ
\bfseries@gt 値を設定できる」という機能が導入されました（元は mweights パッケージの機能）。
\mdseries@mc また、同時に「Computer Modern と Latin Modern の場合は互換性のため太字を
\mdseries@gt bx に、それ以外の欧文ファミリーの場合は太字を b にする」という仕様変更も入りました。これに合わせて、pL^AT_EX 2_ε の和文フォントにも同等の機能を追加し、和文ファミリーの太字も bx ではなく b に変更しました。

```

1888 <|plcore| latexrelease>
1889 \ifx\bfseries@rm\undefined % old
1890 \let\bfseries@mc\undefined
1891 \let\bfseries@gt\undefined
1892 \let\mdseries@mc\undefined
1893 \let\mdseries@gt\undefined
1894 \else % 2020-02-02
1895 \edef\bfseries@mc{\bfdefault}% b
1896 \edef\bfseries@gt{\bfdefault}% b
1897 \edef\mdseries@mc{\mddefault}% m
1898 \edef\mdseries@gt{\mddefault}% m
1899 \fi

```

`\expand@font@defaults` ファミリのデフォルトを完全展開します。まず、オリジナルの L^AT_EX の定義 (l^AT_EXssini.dtx 2020/08/21 v3.2b 以降) を載せておきます。

```

1900 %\def\expand@font@defaults{%
1901 % \edef\rmdef@ult{\rmdefault}%
1902 % \edef\sfddef@ult{\sfdefault}%
1903 % \edef\ttdef@ult{\ttdefault}%
1904 % \series@maybe@drop@one@m\bfdefault\bfdef@ult % !! changed 2020/02/25 v3.1j
1905 % \series@maybe@drop@one@m\mddefault\mddef@ult % !! changed 2020/02/25 v3.1j
1906 % %\edef\famdef@ult{\familydefault}% !! deleted 2020/04/13 v3.1n
1907 % %\@expandfontdefaultshook % !! only in 2020/04/06 v3.1m
1908 % \UseHook{expand@font@defaults}% !! new in 2020/08/21 v3.2b
1909 %}

```

pL^AT_EX では、以下のコードを末尾に追加します。l^AT_EXssini.dtx 2020/04/13 v3.1n で latex3/latex2e#315 対策が入りましたので、その前後で `\expand@font@defaults` および `\init@series@setup` への追加内容が変わります。

```

1910 \ifx\expand@font@defaults\undefined\else %<2020-02-02|2020-10-01|. >
1911 \ifnum\pltx@newhook@avail=\z@ % --- for == 2020-02-02 BEGIN
1912 \g@addto@macro\expand@font@defaults{%
1913 \edef\mcdef@ult{\mcdefault}%
1914 \edef\gtdef@ult{\gtdefault}%
1915 \edef\kanjidef@ult{\kanjifamilydefault}%
1916 }
1917 \else % --- for == 2020-02-02 END & for >= 2020-10-01 BEGIN
1918 \AddToHook{expand@font@defaults}{%
1919 \edef\mcdef@ult{\mcdefault}%
1920 \edef\gtdef@ult{\gtdefault}%
1921 %\edef\kanjidef@ult{\kanjifamilydefault}% !! sync with 2020/04/13 v3.1n
1922 }
1923 \fi % --- for >= 2020-10-01 END
1924 \fi %</2020-02-02|2020-10-01|. >

```

`\bfseries` ファミリーごとの設定値を参照します。まず、オリジナルの L^AT_EX の定義 (l^AT_EXssini.dtx `\mdseries` 2020/09/30 v3.2d 以降) を載せておきます。

```

1925 %\DeclareRobustCommand\bfseries{%
1926 % \not@math@alphabet\bfseries\mathbf
1927 % \expand@font@defaults
1928 % \ifx\bfdefault\bfdefault@previous\else % new in 2020/03/19 v3.1k
1929 % \expandafter\def\expandafter\bfdefault

```



```

1930 % \expandafter{\bfdefault\@empty}%
1931 % \let\bfdefault@previous\bfdefault % bugfix in 2020/09/30 v3.2d
1932 % \let\bfseries@rm\bfdef@ult
1933 % \let\bfseries@sf\bfdef@ult
1934 % \let\bfseries@tt\bfdef@ult
1935 % %\@setbfseriesdefaultshook % !! only in 2020/04/06 v3.1m
1936 % \UseHook{bfseries/defaults}% !! new in 2020/08/21 v3.2b
1937 % \fi
1938 % \ifx\f@family\rmdef@ult \fontseries\bfseries@rm
1939 % \else\ifx\f@family\sdef@ult \fontseries\bfseries@sf
1940 % \else\ifx\f@family\ttdef@ult \fontseries\bfseries@tt
1941 % \else \fontseries\bfdefault
1942 % \fi\fi\fi
1943 % \UseHook{bfseries}% !! new in 2020/08/21 v3.2b
1944 % \selectfont
1945 %}
1946 %\DeclareRobustCommand\mdseries{%
1947 % \not@math@alphabet\mdseries\relax
1948 % \expand@font@defaults
1949 % \ifx\mddefault\mddefault@previous\else % new in 2020/03/19 v3.1k
1950 % \expandafter\def\expandafter\mddefault
1951 % \expandafter{\mddefault\@empty}%
1952 % \let\mddefault@previous\mddefault % bugfix in 2020/09/30 v3.2d
1953 % \let\mdseries@rm\mddef@ult
1954 % \let\mdseries@sf\mddef@ult
1955 % \let\mdseries@tt\mddef@ult
1956 % %\@setmdseriesdefaultshook % !! only in 2020/04/06 v3.1m
1957 % \UseHook{mdseries/defaults}% !! new in 2020/08/21 v3.2b
1958 % \fi
1959 % \ifx\f@family\rmdef@ult \fontseries\mdseries@rm
1960 % \else\ifx\f@family\sdef@ult \fontseries\mdseries@sf
1961 % \else\ifx\f@family\ttdef@ult \fontseries\mdseries@tt
1962 % \else \fontseries\mddefault
1963 % \fi\fi\fi
1964 % \UseHook{mdseries}% !! new in 2020/08/21 v3.2b
1965 % \selectfont
1966 %}

```

以下で p \LaTeX 用に再定義します。まず \LaTeX 2 ϵ 2020-02-02 ベース。

```

1967 \ifx\bfseries@rm\@undefined\else %<*2020-02-02|2020-10-01|. >
1968 \ifnum\pltx@newhook@avail=\z@ % --- for == 2020-02-02 BEGIN
1969 \DeclareRobustCommand\bfseries{%
1970 \not@math@alphabet\bfseries\mathbf
1971 \expand@font@defaults
1972 % changed \fontseries -> \romanseries
1973 \ifx\f@family\rmdef@ult \romanseries\bfseries@rm
1974 \else\ifx\f@family\sdef@ult \romanseries\bfseries@sf
1975 \else\ifx\f@family\ttdef@ult \romanseries\bfseries@tt
1976 \else \romanseries\bfdefault
1977 \fi\fi\fi

```

ここからが p \LaTeX による追加コードです。

```

1978 % changed \fontseries -> \kanjiseries
1979 \ifx\k@family\mcdef@ult \kanjiseries\bfseries@mc

```

```

1980 \else\ifx\k@family\gtdef@ult \kanjiseriess\bfseries@gt
1981 \else \kanjiseriess\bfdefault
1982 \fi\fi

```

ここまで。

```

1983 \selectfont
1984 }
1985 \DeclareRobustCommand\mdseries{%
1986 \not@math@alphabet\mdseries\relax
1987 \expand@font@defaults
1988 % changed \fontseries -> \romanseries
1989 \ifx\f@family\rmdef@ult \romanseries\mdseries@rm
1990 \else\ifx\f@family\sdef@ult \romanseries\mdseries@sf
1991 \else\ifx\f@family\ttdef@ult \romanseries\mdseries@tt
1992 \else \romanseries\mddefault
1993 \fi\fi\fi

```

ここからが p_{La}T_EX による追加コードです。

```

1994 % changed \fontseries -> \kanjiseriess
1995 \ifx\k@family\mcdef@ult \kanjiseriess\mdseries@mc
1996 \else\ifx\k@family\gtdef@ult \kanjiseriess\mdseries@gt
1997 \else \kanjiseriess\mddefault
1998 \fi\fi

```

ここまで。

```

1999 \selectfont
2000 }

```

次に L_AT_EX 2_ε 2020-10-01 ベース。 \AddToHook で十分です。

```

2001 \else % --- for == 2020-02-02 END & for >= 2020-10-01 BEGIN
2002 \AddToHook{bfseries/defaults}{%
2003 \let\bfseries@mc\bfdef@ult
2004 \let\bfseries@gt\bfdef@ult
2005 }
2006 \AddToHook{bfseries}{%
2007 % changed \fontseries -> \kanjiseriess
2008 \ifx\k@family\mcdef@ult \kanjiseriess\bfseries@mc
2009 \else\ifx\k@family\gtdef@ult \kanjiseriess\bfseries@gt
2010 \else \kanjiseriess\bfdefault
2011 \fi\fi
2012 }
2013 \AddToHook{mdseries/defaults}{%
2014 \let\mdseries@mc\mddef@ult
2015 \let\mdseries@gt\mddef@ult
2016 }
2017 \AddToHook{mdseries}{%
2018 % changed \fontseries -> \kanjiseriess
2019 \ifx\k@family\mcdef@ult \kanjiseriess\mdseries@mc
2020 \else\ifx\k@family\gtdef@ult \kanjiseriess\mdseries@gt
2021 \else \kanjiseriess\mddefault
2022 \fi\fi
2023 }
2024 \fi % --- for >= 2020-10-01 END
2025 \fi %</2020-02-02|2020-10-01|. >

```

```

are@family@series@update@kanji \prepare@family@series@update の和文版です。
\@meta@family@list@kanji 2026 \ifx\prepare@family@series@update\@undefined % old
date@series@target@value@kanji 2027 \let\prepare@family@series@update@kanji\@undefined
2028 \let\@meta@family@list@kanji\@undefined
2029 \let\update@series@target@value@kanji\@undefined
2030 \else % 2020-02-02
2031 \def\prepare@family@series@update#1#2{%
2032 ~\if@forced@series
2033 <+debug> \series@change@debug{No series preparation (forced \f@series)\on@line}%
2034 ~ \romanfamily#2% % changed \fontfamily -> \romanfamily
2035 ~\else
2036 <+debug> \series@change@debug{Preparing for switching to #1 (#2)\on@line}%
2037 \expand@font@defaults
2038 \let\target@series@value\@empty
2039 \def\target@meta@family@value{#1}%
2040 \expandafter\edef\csname ??def@ult\endcsname{\f@family}%
2041 \let\@elt@update@series@target@value
2042 \@meta@family@list
2043 \@elt{??}%
2044 \let\@elt\relax
2045 \romanfamily#2% % changed \fontfamily -> \romanfamily
2046 \ifx\target@series@value\@empty
2047 <+debug> \series@change@debug{Target series still empty ...}%
2048 \else
2049 \ifx \f@series\target@series@value
2050 <+debug> \series@change@debug{Target series unchanged:
2051 <+debug> \f@series \space = \target@series@value}%
2052 \else
2053 \maybe@load@fontshape
2054 <+debug> \series@change@debug{Target series:
2055 <+debug> \f@series \space -> \target@series@value}%
2056 % \let\f@series\target@series@value
2057 \series@maybe@drop@one@m\target@series@value\f@series
2058 \fi
2059 \fi
2060 ~\fi
2061 }
2062 \def\prepare@family@series@update@kanji#1#2{%
2063 ~\if@forced@series@kanji
2064 <+debug> \series@change@debug{No series preparation (forced \k@series)\on@line}%
2065 ~ \kanjifamily#2%
2066 ~\else
2067 <+debug> \series@change@debug{Preparing for switching to #1 (#2)\on@line}%
2068 \expand@font@defaults
2069 \let\target@series@value\@empty
2070 \def\target@meta@family@value{#1}%
2071 \expandafter\edef\csname ??def@ult\endcsname{\k@family}%
2072 \let\@elt@update@series@target@value@kanji
2073 \@meta@family@list@kanji
2074 \@elt{??}%
2075 \let\@elt\relax
2076 \kanjifamily#2%
2077 \ifx\target@series@value\@empty
2078 <+debug> \series@change@debug{Target series still empty ...}%

```

```

2079 \else
2080 \ifx \k@series\target@series@value
2081 <+debug> \series@change@debug{Target series unchanged:
2082 <+debug> \k@series \space = \target@series@value}%
2083 \else
2084 \begingroup\let\f@encoding\k@encoding\let\f@family\k@family
2085 \maybe@load@fontshape@endgroup
2086 <+debug> \series@change@debug{Target series:
2087 <+debug> \k@series \space -> \target@series@value}%
2088 % \let\k@series\target@series@value
2089 \series@maybe@drop@one@m\target@series@value\k@series
2090 \fi
2091 \fi
2092 ~\fi
2093 }
2094 \def\@meta@family@list@kanji{\@elt{mc}\@elt{gt}}
2095 \def@update@series@target@value@kanji#1{%
2096 \def\reserved@a{#1}%
2097 \ifx\target@meta@family@value\reserved@a % rm -> rm do nothing
2098 \else
2099 <+debug> \series@change@debug{Trying to match #1: \csname#1def@ult\endcsname
2100 <+debug> \space = \k@family\space ?}%
2101 \expandafter\ifx\csname#1def@ult\endcsname\k@family
2102 \let\@elt@gobble
2103 \expandafter\let\expandafter\reserved@b
2104 \csname mdseries@\target@meta@family@value\endcsname
2105 \expandafter\let\expandafter\reserved@c
2106 \csname bfseries@\target@meta@family@value\endcsname
2107 <+debug> \series@change@debug{Targets for mdseries and bfseries:
2108 <+debug> \reserved@b\space and \reserved@c}%
2109 \expandafter\series@maybe@drop@one@m
2110 \csname mdseries@#1\endcsname\reserved@d
2111 \ifx\reserved@d\k@series
2112 <+debug> \series@change@debug{mdseries@#1 matched -> \reserved@b}%
2113 \let\target@series@value\reserved@b
2114 \else
2115 \expandafter\series@maybe@drop@one@m
2116 \csname bfseries@#1\endcsname\reserved@d
2117 \ifx\reserved@d\k@series
2118 <+debug> \series@change@debug{bfseries@#1 matched -> \reserved@c}%
2119 \let\target@series@value\reserved@c
2120 \else\ifx\k@series\mddef@ult \let\target@series@value\reserved@b
2121 <+debug> \series@change@debug{mddef@ult matched -> \reserved@b}%
2122 \else\ifx\k@series\bfdef@ult \let\target@series@value\reserved@c
2123 <+debug> \series@change@debug{bfdef@ult matched -> \reserved@c}%
2124 \fi\fi\fi\fi
2125 \fi
2126 \fi
2127 }
2128 \fi

```

\init@series@setup \begin{document}で実行される初期化です。まず、オリジナルの L^AT_EX の定義 (ltfssini.dtx 2020/04/13 v3.1n 以降) を載せておきます。

```

2129 %\def\init@series@setup{%
2130 % \ifx\bfseries@rm@kernel\bfseries@rm
2131 % \expandafter\in@\expandafter{\rmdefault}%
2132 % {cmr,cmss,cmtt,lcms,lcmtt,lmr,lmss,lmtt}%
2133 % \ifin@ \else \def\bfseries@rm{b}\fi\fi
2134 % \ifx\bfseries@sf@kernel\bfseries@sf
2135 % \expandafter\in@\expandafter{\sfdefault}%
2136 % {cmr,cmss,cmtt,lcms,lcmtt,lmr,lmss,lmtt}%
2137 % \ifin@ \else \def\bfseries@sf{b}\fi\fi
2138 % \ifx\bfseries@tt@kernel\bfseries@tt
2139 % \expandafter\in@\expandafter{\ttdefault}%
2140 % {cmr,cmss,cmtt,lcms,lcmtt,lmr,lmss,lmtt}%
2141 % \ifin@ \else \def\bfseries@tt{b}\fi\fi
2142 % %\expand@font@defaults % !! deleted in 2020/04/13 v3.1n BEGIN
2143 % %\ifx\famdef@ult\rmdef@ult \rmfamily % !! CONT
2144 % %\else\ifx\famdef@ult\sfdef@ult \sffamily % !! CONT
2145 % %\else\ifx\famdef@ult\ttdef@ult \ttfamily % !! CONT
2146 % %\fi\fi\fi % !! deleted in 2020/04/13 v3.1n END
2147 % \reset@font % !! added in 2020/04/13 v3.1n BEGIN
2148 % \ifx\seriesdefault\seriesdefault@kernel % !! CONT
2149 % \mdseries % !! CONT
2150 % \let\seriesdefault\fontseries % !! CONT
2151 % \fi % !! added in 2020/04/13 v3.1n END
2152 %}%

```

ここからが pL^AT_EX による追加コードです。

- L^AT_EX 2_ε 2019-10-01 以前：未定義
- L^AT_EX 2_ε 2020-02-02 以降：上のおりの定義
- ただし、latexrelease で巻き戻し：`\relax` と同義

になることに注意します。

```

2153 \expandafter\ifx\csname init@series@setup\endcsname\relax
2154 \else %<2020-02-02|2020-10-01|. >
2155 \ifnum\pltx@newhook@avail=\z@ % --- for == 2020-02-02 BEGIN
2156 \g@addto@macro\init@series@setup{%
2157 \ifx\kanjidef@ult\mcdef@ult \mcfamily
2158 \else\ifx\kanjidef@ult\gtdef@ult \gtfamily
2159 \fi\fi
2160 }%
2161 \else % --- for == 2020-02-02 END & for >= 2020-10-01 BEGIN
2162 \g@addto@macro\init@series@setup{%
2163 % !! sync with 2020/04/13 v3.1n BEGIN
2164 \ifx\kanjiseriessdefault\kanjiseriessdefault@kernel
2165 \mdseries
2166 \let\kanjiseriessdefault\k@series
2167 \fi
2168 % !! sync with 2020/04/13 v3.1n END
2169 }%
2170 \fi % --- for >= 2020-10-01 END
2171 \fi %</2020-02-02|2020-10-01|. >

```

`\kanjiseriessdefault` `\kanjiseriessdefault` が `pldefs.ltx` または `pldefs.cfg` で定義された後に、`\kanjiseriessdefault@kernel` の末尾に `\@empty` を追加します。これは展開時に消滅しますが、文書のプリアンブルで別の値に変更されたかどうか検知できるようになります。

```
2172 \ifnum\pltx@newhook@avail>\z@ % --- for >= 2020-10-01 BEGIN
2173 \def\code@after@pldefs{%
2174 \expandafter\def\expandafter\kanjiseriessdefault
2175 \expandafter{\kanjiseriessdefault\@empty}
2176 \let\kanjiseriessdefault@kernel\kanjiseriessdefault}
2177 \fi % --- for >= 2020-10-01 END
```

`\mcfamily` 和文書体を明朝体にする `\mcfamily` とゴシック体にする `\gtfamily` を定義します。
`\gtfamily` これらは、`\rmfamily` などに対応します。`\mathmc` と `\mathgt` は数式内で用いるときのコマンド名です。

```
2178 \ifx\prepare@family@series@update@kanji\@undefined % old
2179 \DeclareRobustCommand\mcfamily
2180 {\not@math@alphabet\mcfamily\mathmc
2181 \kanjifamily\mcdefault\selectfont}
2182 \DeclareRobustCommand\gtfamily
2183 {\not@math@alphabet\gtfamily\mathgt
2184 \kanjifamily\gtdefault\selectfont}
2185 \else % 2020-02-02
2186 \DeclareRobustCommand\mcfamily
2187 {\not@math@alphabet\mcfamily\mathmc
2188 \prepare@family@series@update@kanji{mc}\mcdefault\selectfont}
2189 \DeclareRobustCommand\gtfamily
2190 {\not@math@alphabet\gtfamily\mathgt
2191 \prepare@family@series@update@kanji{gt}\gtdefault\selectfont}
2192 \fi
2193 \</plcore | latexrelease>
```

`\textmc` テキストファミリを切り替えるためのコマンドです。`lftntcmd.dtx` で定義されて
`\textgt` いる `\textrm` などに対応します。

```
2194 \<plcore>
2195 \DeclareTextFontCommand{\textmc}{\mcfamily}
2196 \DeclareTextFontCommand{\textgt}{\gtfamily}
2197 \</plcore>
```

後回しにしていた他のバージョンへの対処です。ここで新 NFSS 対応コードが終わりますので、`\catcode` トリックを元に戻します。

```
2198 \<plcore | latexrelease>
2199 %%
2200 \ifnum\pltx@latex@level>0\relax % 2020-02-02
2201 %
2202 \ifnum\pltx@latex@level<3\relax % 2020-02-02 patch level 0--2 (no flags)
2203 \DeclareRobustCommand\romanseries[1]{\merge@font@series{#1}}
2204 \DeclareRobustCommand\kanjiseriess[1]{\merge@kanji@series{#1}}
2205 \DeclareRobustCommand\fontseries[1]{\kanjiseriess{#1}\romanseries{#1}}
2206 \DeclareRobustCommand\romanseriesforce[1]{\edef\f@series{#1}}
2207 \DeclareRobustCommand\kanjiseriessforce[1]{\edef\k@series{#1}}
2208 \DeclareRobustCommand\fontseriesforce[1]{\kanjiseriessforce{#1}\romanseriesforce{#1}}
```

```

2209 \fi
2210 %
2211 \ifnum\pltx@latex@level=1\relax % 2020-02-02 patch level 0 (\@reserveda)
2212 \def\merge@kanji@series@#1#2#3\@nil{%
2213   \def\@reserveda{#3}%
2214   \ifx\@reserveda\@empty
2215     \set@target@series@kanji{#2}%
2216   \else
2217     \begingroup\let\f@encoding\k@encoding\let\f@family\k@family
2218     \maybe@load@fontshape@endgroup
2219     \edef\@reserveda{\k@encoding /\k@family /#1/\k@shape}%
2220     \ifcsname \@reserveda \endcsname
2221       \set@target@series@kanji{#1}%
2222     \else
2223       \ifcsname \k@encoding /\k@family /#2/\k@shape \endcsname
2224         \set@target@series@kanji{#2}%
2225         {\let\curr@fontshape\curr@kfontshape\@font@shape@subst@warning}%
2226       \else
2227         \set@target@series@kanji{#3}%
2228         {\let\curr@fontshape\curr@kfontshape\@font@shape@subst@warning}%
2229     \fi
2230   \fi
2231 \fi
2232 }
2233 \def\merge@kanji@shape@#1#2#3\@nil{%
2234   \def\@reserveda{#3}%
2235   \ifx\@reserveda\@empty
2236     \edef\k@shape{#2}%
2237   \else
2238     \begingroup\let\f@encoding\k@encoding\let\f@family\k@family
2239     \maybe@load@fontshape@endgroup
2240     \edef\@reserveda{\k@encoding /\k@family /\k@series/#1}%
2241     \ifcsname \@reserveda \endcsname
2242       \edef\k@shape{#1}%
2243     \else
2244       \ifcsname \k@encoding /\k@family /\k@series/#2 \endcsname
2245         \edef\k@shape{#2}%
2246         {\let\curr@fontshape\curr@kfontshape\@font@shape@subst@warning}%
2247       \else
2248         \edef\k@shape{#3}%
2249         {\let\curr@fontshape\curr@kfontshape\@font@shape@subst@warning}%
2250     \fi
2251   \fi
2252 \fi
2253 }
2254 \fi
2255 %
2256 \ifnum\pltx@latex@level<4\relax % 2020-02-02 patch level 0--4 (drop m)
2257 \def\set@target@series@kanji#1{%
2258   \edef\k@series{#1}%
2259   \edef\k@series{\expandafter\series@drop@one@m\k@series mm\series@drop@one@m}%
2260 }
2261 \else\ifnum\pltx@latex@level=4\relax % 2020-02-02 patch level 5 (old syntax)
2262 \def\set@target@series@kanji#1{%

```

```

2263 \edef\k@series{#1}%
2264 \expandafter\series@maybe@drop@one@m\expandafter{\k@series}\k@series
2265 }
2266 \fi\fi
2267 %
2268 \ifnum\pltx@latex@level<5\relax % 2020-02-02 patch level 0--5
2269 \def\prepare@family@series@update#1#2{%
2270 ~\if@forced@series
2271 (+debug) \series@change@debug{No series preparation (forced \f@series)\on@line}%
2272 ~ \romanfamily#2% % changed \fontfamily -> \romanfamily
2273 ~\else
2274 (+debug) \series@change@debug{Preparing for switching to #1 (#2)\on@line}%
2275 \expand@font@defaults
2276 \let\target@series@value\@empty
2277 \def\target@meta@family@value{#1}%
2278 ~ \expandafter\edef\csname ??def@ult\endcsname{\f@family}%
2279 \let\@elt@update@series@target@value
2280 \@meta@family@list
2281 ~ \@elt{??}%
2282 \let\@elt\relax
2283 \romanfamily#2% % changed \fontfamily -> \romanfamily
2284 \ifx\target@series@value\@empty
2285 (+debug) \series@change@debug{Target series still empty ...}%
2286 \else
2287 \ifx \f@series\target@series@value
2288 (+debug) \series@change@debug{Target series unchanged:
2289 (+debug) \f@series \space = \target@series@value}%
2290 \else
2291 \maybe@load@fontshape
2292 (+debug) \series@change@debug{Target series:
2293 (+debug) \f@series \space -> \target@series@value}%
2294 \let\f@series\target@series@value
2295 \fi
2296 \fi
2297 ~\fi
2298 }
2299 \def\prepare@family@series@update@kanji#1#2{%
2300 ~\if@forced@series@kanji
2301 (+debug) \series@change@debug{No series preparation (forced \k@series)\on@line}%
2302 ~ \kanjifamily#2%
2303 ~\else
2304 (+debug) \series@change@debug{Preparing for switching to #1 (#2)\on@line}%
2305 \expand@font@defaults
2306 \let\target@series@value\@empty
2307 \def\target@meta@family@value{#1}%
2308 ~ \expandafter\edef\csname ??def@ult\endcsname{\k@family}%
2309 \let\@elt@update@series@target@value@kanji
2310 \@meta@family@list@kanji
2311 ~ \@elt{??}%
2312 \let\@elt\relax
2313 \kanjifamily#2%
2314 \ifx\target@series@value\@empty
2315 (+debug) \series@change@debug{Target series still empty ...}%
2316 \else

```



```

2317 \ifx \k@series\target@series@value
2318 <+debug> \series@change@debug{Target series unchanged:
2319 <+debug> \k@series \space = \target@series@value}%
2320 \else
2321 \begingroup\let\f@encoding\k@encoding\let\f@family\k@family
2322 \maybe@load@fontshape@endgroup
2323 <+debug> \series@change@debug{Target series:
2324 <+debug> \k@series \space -> \target@series@value}%
2325 \let\k@series\target@series@value
2326 \fi
2327 \fi
2328 ~\fi
2329 }
2330 \def\@meta@family@list@kanji{\@elt{mc}\@elt{gt}}
2331 \def\update@series@target@value@kanji#1{%
2332 \def\reserved@a{#1}%
2333 \ifx\target@meta@family@value\reserved@a % rm -> rm do nothing
2334 \else
2335 <+debug> \series@change@debug{Trying to match #1: \csname#1def\ult\endcsname
2336 <+debug> \space = \k@family\space ?}%
2337 \expandafter\ifx\csname#1def\ult\endcsname\k@family
2338 \let\@elt@gobble
2339 \expandafter\let\expandafter\reserved@b
2340 \csname mdseries@\target@meta@family@value\endcsname
2341 \expandafter\let\expandafter\reserved@c
2342 \csname bfseries@\target@meta@family@value\endcsname
2343 <+debug> \series@change@debug{Targets for mdseries and bfseries:
2344 <+debug> \reserved@b\space and \reserved@c}%
2345 \expandafter\ifx\csname mdseries@#1\endcsname\k@series
2346 <+debug> \series@change@debug{mdseries@#1 matched -> \reserved@b}%
2347 \let\target@series@value\reserved@b
2348 \else\expandafter\ifx\csname bfseries@#1\endcsname\k@series
2349 <+debug> \series@change@debug{bfseries@#1 matched -> \reserved@c}%
2350 \let\target@series@value\reserved@c
2351 \else\ifx\k@series\mddef\ult \let\target@series@value\reserved@b
2352 <+debug> \series@change@debug{mddef\ult matched -> \reserved@b}%
2353 \else\ifx\k@series\bfdef\ult \let\target@series@value\reserved@c
2354 <+debug> \series@change@debug{bfdef\ult matched -> \reserved@c}%
2355 \fi\fi\fi\fi
2356 \fi
2357 \fi
2358 }
2359 \fi
2360 %
2361 \fi
2362 %%
2363 \pltx@reset@catcode@trick
2364 </plcore|platexrelease>

```

\romanprocess@table 文書の先頭で、和文デフォルトフォントの変更が反映されないのを修正します。

```

\kanjiprocess@table365 (*plcore)
\process@table366 \let\romanprocess@table\process@table
2367 \def\kanjiprocess@table{%
2368 \kanjiencoding\kanjiencodingdefault

```

```

2369 \edef\k@family{\kanjifamilydefault}%
2370 \edef\k@series{\kanjiseriessdefault}%
2371 \edef\k@shape{\kanjishapedefault}%
2372 }
2373 \def\process@table{%
2374   \romanprocess@table
2375   \kanjiprocess@table
2376 }
2377 \@onlypreamble\romanprocess@table
2378 \@onlypreamble\kanjiprocess@table
2379 </plcore>

```

7.3 強調書体

`\em` 従来は `\em`, `\emph` で和文フォントの切り替えは行っていませんでしたが、和文フォントも `\gtfamily` に切り替えるようにしました。

`\emminersshape` [pLaTeX 2_ε 2016/04/17] LaTeX <2015/01/01>で追加された `\emminersshape` も取り入れ、強調コマンドを入れ子にする場合の書体を自由に再定義できるようになりました。

[pLaTeX 2_ε 2020-02-02] LaTeX <2020-02-02>で追加された `\DeclareEmphSequence` をサポートしました。

```

2380 <latexrelease>\plIncludeInRelease{2020/02/02}{\DeclareEmphSequence}
2381 <latexrelease>                                {Nested emph}%
2382 <*plcore | latexrelease>
2383 \ifx\DeclareEmphSequence\undefined % old
2384 \DeclareRobustCommand\em
2385   {\@nomath\em \ifdim \fontdimen\@ne\font >\z@
2386     \emminersshape \else \gtfamily \itshape \fi}%
2387 \else
2388 \DeclareRobustCommand\em{%                % 2020-02-02
2389   \@nomath\em
2390   \ifx\emfontdeclare@clist\empty
2391     \ifdim \fontdimen\@ne\font >\z@
2392       \emminersshape \else \gtfamily \itshape \fi
2393   \else
2394     \edef\em@currfont{\csname\curr@fontshape/\f@size\endcsname}%
2395     \expandafter\do@emfont@update\emfontdeclare@clist\do@emfont@update
2396   \fi
2397 }
2398 \fi
2399 \def\emminersshape{\mcfamily \upshape}%
2400 </plcore | latexrelease>
2401 <latexrelease>\plEndIncludeInRelease
2402 <latexrelease>\plIncludeInRelease{2016/04/17}{\DeclareEmphSequence}
2403 <latexrelease>                                {Support \emminersshape}%
2404 <latexrelease>\DeclareRobustCommand\em
2405 <latexrelease>    {\@nomath\em \ifdim \fontdimen\@ne\font >\z@
2406 <latexrelease>                                \emminersshape \else \gtfamily \itshape \fi}%
2407 <latexrelease>\def\emminersshape{\mcfamily \upshape}%
2408 <latexrelease>\plEndIncludeInRelease
2409 <latexrelease>\plIncludeInRelease{2015/01/01}{\DeclareEmphSequence}

```

```

2410 <platexrelease>                                     {Non-supported \eminnershape}%
2411 <platexrelease>\DeclareRobustCommand\em
2412 <platexrelease>          {\@nomath\em \ifdim \fontdimen\@ne\font >\z@
2413 <platexrelease>                                     \mcfamily \upshape \else \gtfamily \itshape \fi}
2414 <platexrelease>\def\eminnershape{\upshape}% defined by LaTeX, but not used by pLaTeX
2415 <platexrelease>\plEndIncludeInRelease
2416 <platexrelease>\plIncludeInRelease{0000/00/00}{\DeclareEmphSequence}
2417 <platexrelease>                                     {ASCII Corporation original}%
2418 <platexrelease>\DeclareRobustCommand\em
2419 <platexrelease>          {\@nomath\em \ifdim \fontdimen\@ne\font >\z@
2420 <platexrelease>                                     \mcfamily \upshape \else \gtfamily \itshape \fi}
2421 <platexrelease>\let\eminnershape\@undefined
2422 <platexrelease>\plEndIncludeInRelease

```

7.4 下線マクロ

`\textunderscore` このコマンドはテキストモードで指定された `_` の内部コマンドです。縦組での位置を調整するように再定義をします。もとは `ltoutenc.dtx` で定義されています。

なお、`_` を数式モードで使うと `\mathunderscore` が実行されます。

コミュニティ版では縦数式ディレクションでベースライン補正量が変わったのを直しました。あわせて横ディレクションでもベースライン補正に追従するようにしています。

```

2423 <platexrelease>\plIncludeInRelease{2017/04/08}{\textunderscore}
2424 <platexrelease>                                     {Baseline shift for \textunderscore}%
2425 <*plcore | platexrelease>
2426 \DeclareTextCommandDefault{\textunderscore}{%
2427   \leavevmode\kern.06em
2428   \raise-\iftdir\ifmdir\ybaselineshift
2429     \else\tbaselineshift\fi
2430     \else\ybaselineshift\fi
2431   \vbox{\hrule\@width.3em}}
2432 </plcore | platexrelease>
2433 <platexrelease>\plEndIncludeInRelease
2434 <platexrelease>\plIncludeInRelease{0000/00/00}{\textunderscore}
2435 <platexrelease>                                     {ASCII Corporation original}%
2436 <platexrelease>\DeclareTextCommandDefault{\textunderscore}{%
2437   \leavevmode\kern.06em
2438   \iftdir\raise-\tbaselineshift\fi
2439   \vbox{\hrule\@width.3em}}
2440 <platexrelease>\plEndIncludeInRelease

```

7.5 合成文字

L^AT_EX 2_ε のカーネルのコードをそのまま使うと、p_TE_X のベースライン補正量がゼロでないときに合成文字がおかしくなっていたため、対策します。

`\pltx@saved@oalign` `\b{...}`, `\c{...}`, `\d{...}`, `\k{...}`などの合成文字を修正するため、`ltplain.dtx` の `\oalign` を上書きします。

```

2441 <platexrelease>%\plIncludeInRelease{0000/00/00}{\pltx@saved@oalign}
2442 <platexrelease>%    {Special case! (This block is required for any emulation date)}%

```

```
2443 <*plcore | platexrelease>
```

まず、元の L^AT_EX のコードをコピーしたものです。接頭辞 `\pltx@saved...` を付けておきます。

```
2444 \def\pltx@saved@oalign#1{\leavevmode\top{\baselineskip\z@skip \lineskip.25ex%
2445   \ialign{##\crr#1\crr}}}%
2446 </plcore | platexrelease>
2447 <platexrelease>%\plEndIncludeInRelease
```

`\pltx@oalign` 次に、pL^AT_EX の新しいコードです。

```
2448 <platexrelease>\plIncludeInRelease{2018/07/28}{\pltx@oalign}
2449 <platexrelease>           {Fix for non-zero baselineshift}}%
2450 <*plcore | platexrelease>
2451 \def\pltx@oalign#1{\ifmmode
2452   \leavevmode\top{\baselineskip\z@skip \lineskip.25ex%
2453   \ialign{##\crr#1\crr}}}%
2454 \else
2455   \iftdir\ybaselineshift\tbaselineshift\fi
2456   \m@th$\hbox{\top{\baselineskip\z@skip \lineskip.25ex%
2457   \ialign{##\crr#1\crr}}}$%
2458 \fi}
2459 </plcore | platexrelease>
2460 <platexrelease>\plEndIncludeInRelease
2461 <platexrelease>\plIncludeInRelease{0000/00/00}{\pltx@oalign}
2462 <platexrelease>           {Fix for non-zero baselineshift}}%
2463 <platexrelease>\let\pltx@oalign\@undefined
2464 <platexrelease>\plEndIncludeInRelease
```

`\pltx@saved@ltx@sh@ft` `\b{...}` と `\d{...}` の合成文字を修正するため、l^AT_EX の `\ltx@sh@ft` を上書きします。

```
2465 <platexrelease>%\plIncludeInRelease{0000/00/00}{\pltx@saved@ltx@sh@ft}
2466 <platexrelease>%   {Special case! (This block is required for any emulation date)}%
2467 <*plcore | platexrelease>
```

まず、元の L^AT_EX のコードをコピーしたものです。接頭辞 `\pltx@saved...` を付けておきます。

```
2468 \def\pltx@saved@ltx@sh@ft #1{%
2469   \dimen@ #1%
2470   \kern \strip@pt
2471   \fontdimen1\font \dimen@
2472   } % kern by #1 times the current slant
2473 </plcore | platexrelease>
2474 <platexrelease>%\plEndIncludeInRelease
```

`\pltx@ltx@sh@ft` 次に、pL^AT_EX の新しいコードです。

```
2475 <platexrelease>\plIncludeInRelease{2018/07/28}{\pltx@ltx@sh@ft}
2476 <platexrelease>           {Fix for non-zero baselineshift}}%
2477 <*plcore | platexrelease>
2478 \def\pltx@ltx@sh@ft #1{%
2479   \ybaselineshift\z@
2480   \dimen@ #1%
2481   \kern \strip@pt
```

```

2482 \fontdimen1\font \dimen@
2483 } % kern by #1 times the current slant
2484 </plcore| platexrelease>
2485 <platexrelease>\plEndIncludeInRelease
2486 <platexrelease>\plIncludeInRelease{0000/00/00}{\pltx@ltx@sh@ft}
2487 <platexrelease> {Fix for non-zero baselineshift}%
2488 <platexrelease>\let\pltx@ltx@sh@ft\@undefined
2489 <platexrelease>\plEndIncludeInRelease

```

`\g@tlastchart@` TeX Live 2015 で追加された `\lastnodechar` を利用して、「直前の文字」の符号位置を得るコードです。`\lastnodechar` が未定義の場合は `-1` が返ります。

```

2490 <platexrelease>\plIncludeInRelease{2016/06/10}{\g@tlastchart@}
2491 <platexrelease> {Added \g@tlastchart@}%
2492 <*plcore| platexrelease>
2493 \def\g@tlastchart@#1{#1\ifx\lastnodechar\@undefined\m@ne\else\lastnodechar\fi}
2494 </plcore| platexrelease>
2495 <platexrelease>\plEndIncludeInRelease
2496 <platexrelease>\plIncludeInRelease{0000/00/00}{\g@tlastchart@}
2497 <platexrelease> {Added \g@tlastchart@}%
2498 <platexrelease>\let\g@tlastchart@\@undefined
2499 <platexrelease>\plEndIncludeInRelease

```

`\pltx@isletter` 第一引数のマクロ (`#1`) の置換テキストが、カテゴリコード 11 か 12 の文字トークン 1 文字であった場合に第二引数の内容に展開され、そうでない場合は第三引数の内容に展開されます。

```

2500 <platexrelease>\plIncludeInRelease{2018/07/28}{\pltx@isletter}
2501 <platexrelease> {Support PD1 encoding}%
2502 <*plcore| platexrelease>
2503 \def\pltx@mark{\pltx@mark@}
2504 \let\pltx@scanstop\relax
2505 \long\def\pltx@cond#1\fi{%
2506   #1\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi}
2507 \def\pltx@pdfencA{PD1}
2508 \def\pltx@composite@chkenc{%
2509   \ifx\pltx@pdfencA\f@encoding
2510     \expandafter\@firstoftwo
2511   \else
2512     \expandafter\@secondoftwo
2513   \fi}
2514 \long\def\pltx@isletter#1{%
2515   \expandafter\pltx@isletter@i#1\pltx@scanstop}
2516 \long\def\pltx@isletter@i#1\pltx@scanstop{%
2517   \pltx@cond\ifx\pltx@mark#1\pltx@mark\fi{\@firstoftwo}%
2518   {\pltx@isletter@ii\pltx@scanstop#1\pltx@scanstop}{#1\pltx@mark}}
2519 \long\def\pltx@isletter@ii#1\pltx@scanstop#1{%
2520   \pltx@cond\ifx\pltx@mark#1\pltx@mark\fi%
2521   {\pltx@isletter@iii}{\pltx@isletter@iv}}
2522 \long\def\pltx@isletter@iii#1\pltx@mark{\@secondoftwo}
2523 \long\def\pltx@isletter@iv#1#2#3\pltx@mark{%
2524   \pltx@cond\ifx\pltx@mark#3\pltx@mark\fi%
2525   \pltx@cond{\ifnum0\ifcat A\noexpand#21\fi\ifcat=\noexpand#21\fi>\z@}\fi
2526   {\@firstoftwo}{\pltx@composite@chkenc}%

```

```

2527 }{\pltx@composite@chkenc}}
2528 /plcore| latexrelease)
2529 <latexrelease>\plEndIncludeInRelease
2530 <latexrelease>\plIncludeInRelease{2016/06/10}{\pltx@isletter}
2531 <latexrelease> {Added \pltx@isletter}%
2532 <latexrelease>\def\pltx@mark{\pltx@mark@}
2533 <latexrelease>\let\pltx@scanstop\relax
2534 <latexrelease>\long\def\pltx@cond#1\fi{%
2535 <latexrelease> #1\expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi}
2536 <latexrelease>\long\def\pltx@isletter#1{%
2537 <latexrelease> \expandafter\pltx@isletter@i#1\pltx@scanstop}
2538 <latexrelease>\long\def\pltx@isletter@i#1\pltx@scanstop{%
2539 <latexrelease> \pltx@cond\ifx\pltx@mark#1\pltx@mark\fi{\@firstoftwo}%
2540 <latexrelease> {\pltx@isletter@ii\pltx@scanstop#1\pltx@scanstop}{#1\pltx@mark}}
2541 <latexrelease>\long\def\pltx@isletter@ii#1\pltx@scanstop{%
2542 <latexrelease> \pltx@cond\ifx\pltx@mark#1\pltx@mark\fi%
2543 <latexrelease> {\pltx@isletter@iii}{\pltx@isletter@iv}}
2544 <latexrelease>\long\def\pltx@isletter@iii#1\pltx@mark{\@secondoftwo}
2545 <latexrelease>\long\def\pltx@isletter@iv#1#2#3\pltx@mark{%
2546 <latexrelease> \pltx@cond\ifx\pltx@mark#3\pltx@mark\fi{%
2547 <latexrelease> \pltx@cond{\ifnum0\ifcat A\noexpand#21\fi\ifcat=\noexpand#21\fi>z@}\fi
2548 <latexrelease> {\@firstoftwo}{\@secondoftwo}%
2549 <latexrelease> }{\@secondoftwo}}
2550 <latexrelease>\plEndIncludeInRelease
2551 <latexrelease>\plIncludeInRelease{0000/00/00}{\pltx@isletter}
2552 <latexrelease> {Added \pltx@isletter}%
2553 <latexrelease>\let\pltx@isletter\undefined
2554 <latexrelease>\plEndIncludeInRelease

```

\@text@composite 合成文字の内部命令です。v1.6a で誤って L^AT_EX の定義を上書きしてしまいました
が、v1.6c で外しました。

```

2555 <latexrelease>\plIncludeInRelease{2016/06/10}{\@text@composite}
2556 <latexrelease> {Fix for non-zero baselineshift (revert)}%
2557 <latexrelease>\def\@text@composite#1#2#3\@text@composite{%
2558 <latexrelease> \expandafter\@text@composite@x
2559 <latexrelease> \csname\string#1-\string#2\endcsname}
2560 <latexrelease>\plEndIncludeInRelease
2561 <latexrelease>\plIncludeInRelease{2016/04/17}{\@text@composite}
2562 <latexrelease> {Fix for non-zero baselineshift (wrong)}%
2563 <latexrelease>\def\@text@composite#1#2#3{%
2564 <latexrelease> \begingroup
2565 <latexrelease> \setbox\z@=\hbox\bgroup%
2566 <latexrelease> \ybaselineshift\z@\tbaselineshift\z@
2567 <latexrelease> \expandafter\@text@composite@x
2568 <latexrelease> \csname\string#1-\string#2\endcsname}
2569 <latexrelease>\plEndIncludeInRelease
2570 <latexrelease>\plIncludeInRelease{0000/00/00}{\@text@composite}
2571 <latexrelease> {LaTeX2e original}%
2572 <latexrelease>\def\@text@composite#1#2#3\@text@composite{%
2573 <latexrelease> \expandafter\@text@composite@x
2574 <latexrelease> \csname\string#1-\string#2\endcsname}
2575 <latexrelease>\plEndIncludeInRelease

```

`\pltx@saved@text@composite@x` 合成文字の内部命令 `\@text@composite@x` のために、2通りの定義を準備します。

```
2576 <latexrelease>%\plIncludeInRelease{0000/00/00}{\pltx@saved@text@composite@x}
2577 <latexrelease>%    {Special case! (This block is required for any emulation date)}%
2578 <*plcore | latexrelease>
```

まず、元の L^AT_EX のコードをコピーしたものです。接頭辞 `\pltx@saved...` を付けておきます。

```
2579 \def\pltx@saved@text@composite@x#1{%
2580     \ifx#1\relax
2581         \expandafter\@secondoftwo
2582     \else
2583         \expandafter\@firstoftwo
2584     \fi
2585     #1}
2586 </plcore | latexrelease>
2587 <latexrelease>%\plEndIncludeInRelease
```

`\pltx@text@composite@x` 次に、pL^AT_EX の新しいコードです。`\g@tlastchart@`と `\pltx@isletter` を使います。

```
2588 <latexrelease>\plIncludeInRelease{2018/07/28}{\pltx@text@composite@x}
2589 <latexrelease>                {Fix for non-zero baselineshift}%
2590 <*plcore | latexrelease>
2591 \def\pltx@text@composite@x#1#2{%
2592     \ifx#1\relax
2593         #2%
2594     \else\pltx@isletter{#1}{#1}{%
2595         \begingroup
```

`#1` を実際に組んでみて、符号位置の取得を試みます。結果は `\@tempcntb` に保存されます。取得に失敗した場合は `-1` です。

```
2596     \setbox\z@\hbox\bgroup
2597         \ybaselineshift\z@\tbaselineshift\z@
2598     #1%
2599     \g@tlastchart@\@tempcntb
2600     \xdef\pltx@composite@temp{\noexpand\@tempcntb=\the\@tempcntb\relax}%
2601     \aftergroup\pltx@composite@temp
2602     \egroup
```

アクセントが付く「本体の文字」が欧文文字と推測される場合には、一旦数式モードに入ることによって `\xkanjiskip` が前後に入るようにします。ここでは、取得に失敗した場合も欧文文字であると仮定しています。また、符号位置の取得に成功していた場合は、その `\xspcode` の状態に応じて、数式モードの前後に `\null` を補って `\xkanjiskip` の挿入を抑制します。

```
2603     \ifnum\@tempcntb<\@cclvi
2604     \ifnum\@tempcntb>\m@ne
2605         \ifodd\xspcode\@tempcntb\else\leavevmode\null\fi
2606     \fi
2607     \begingroup\m@th$%
2608         \ifx\textbaselineshiftfactor\@undefined\else
2609             \textbaselineshiftfactor\z@\fi
2610         \box\z@
```

```

2611      $\endgroup
2612      \ifnum\@tempcntb>\m@ne
2613          \ifnum\xspcode\@tempcntb<2\relax\fi
2614      \fi

```

アクセントが付く「本体の文字」が和文文字と推測される場合には、ベースライン補正を行わずに出力します。

```

2615      \else
2616          {\ybaselineshift\z@\tbaselineshift\z@#1}%
2617      \fi
2618      \endgroup}%
2619      \fi
2620 }
2621 \plcore | latexrelease)
2622 \plEndIncludeInRelease
2623 \plIncludeInRelease{2016/06/10}{\pltx@text@composite@x}
2624 \pltx@text@composite@x{Fix for non-zero baselineshift}%
2625 \def\pltx@text@composite@x#1#2{%
2626     \ifx#1\relax
2627         #2%
2628     \else\pltx@isletter{#1}{#1}{%
2629         \begingroup
2630         \setbox\z@\hbox\bgroup%
2631         \ybaselineshift\z@\tbaselineshift\z@
2632         #1%
2633         \g@tlastchart@\@tempcntb
2634         \xdef\pltx@composite@temp{\noexpand\@tempcntb=\the\@tempcntb\relax}%
2635         \aftergroup\pltx@composite@temp
2636     \egroup
2637     \ifnum\@tempcntb<\z@
2638         \@tempdima=\iftdir
2639             \ifmdir
2640                 \ifmmode\tbaselineshift\else\ybaselineshift\fi
2641             \else
2642                 \tbaselineshift
2643             \fi
2644         \else
2645             \ybaselineshift
2646         \fi
2647         \@tempcntb=\@cclvi
2648     \else\@tempdima=\z@
2649     \fi
2650     \ifnum\@tempcntb<\@cclvi
2651         \ifnum\@tempcntb>\m@ne\ifnum\@tempcntb<\@cclvi
2652             \ifodd\xspcode\@tempcntb\else\leavevmode\hbox{}\fi
2653         \fi\fi
2654         \begingroup\mathsurround\z@{%
2655             \ifx\textbaselineshiftfactor\undefined\else
2656                 \textbaselineshiftfactor\z@\fi
2657             \box\z@
2658         }\endgroup%
2659         \ifnum\@tempcntb>\m@ne\ifnum\@tempcntb<\@cclvi
2660             \ifnum\xspcode\@tempcntb<2\hbox{}\fi
2661         \fi\fi

```



```

2662 <platexrelease> \else
2663 <platexrelease> \ifdim\@tempdima=\z@{\ybaselineshift\z@\tbaselineshift\z@#1}%
2664 <platexrelease> \else\leavevmode\lower\@tempdima\box\z@\fi
2665 <platexrelease> \fi
2666 <platexrelease> \endgroup}%
2667 <platexrelease> \fi
2668 <platexrelease> }
2669 <platexrelease> \plEndIncludeInRelease
2670 <platexrelease> \plIncludeInRelease{2016/04/17}{\pltx@text@composite@x}
2671 <platexrelease> {Fix for non-zero baselineshift}%
2672 <platexrelease> \def\pltx@text@composite@x#1#2{%
2673 <platexrelease> \ifx#1\relax
2674 <platexrelease> \expandafter\@secondoftwo
2675 <platexrelease> \else
2676 <platexrelease> \expandafter\@firstoftwo
2677 <platexrelease> \fi
2678 <platexrelease> #1#2}\egroup
2679 <platexrelease> \leavevmode
2680 <platexrelease> \expandafter\lower
2681 <platexrelease> \iftdir
2682 <platexrelease> \ifmdir
2683 <platexrelease> \ifmmode\tbaselineshift\else\ybaselineshift\fi
2684 <platexrelease> \else
2685 <platexrelease> \tbaselineshift
2686 <platexrelease> \fi
2687 <platexrelease> \else
2688 <platexrelease> \ybaselineshift
2689 <platexrelease> \fi
2690 <platexrelease> \box\z@
2691 <platexrelease> \endgroup}
2692 <platexrelease> \plEndIncludeInRelease
2693 <platexrelease> \plIncludeInRelease{0000/00/00}{\pltx@text@composite@x}
2694 <platexrelease> {Fix for non-zero baselineshift}%
2695 <platexrelease> \let\pltx@text@composite@x\@undefined
2696 <platexrelease> \plEndIncludeInRelease

```

\fixcompositeaccent 上記 2 通りの定義のうち、本当は pL^AT_EX の定義を用いたのですが、想定外の
\nofixcompositeaccent エラーが発生するのを防ぐため、デフォルトでは L^AT_EX の定義のままとしておき
\@text@composite@x ます。そして、\fixcompositeaccent が有効な時だけ pL^AT_EX の定義を用います。

\nofixcompositeaccent はこの否定です。

```

2697 <platexrelease> %\plIncludeInRelease{0000/00/00}{\@text@composite@x}
2698 <platexrelease> % {Special case! (This block is required for any emulation date)}%
2699 <platexrelease> *| core | platexrelease)
2700 \DeclareRobustCommand\fixcompositeaccent{%
2701 \let\oalign\pltx@oalign
2702 \let\ltx@sh@ft\pltx@ltx@sh@ft
2703 \let\@text@composite@x\pltx@text@composite@x
2704 }
2705 \DeclareRobustCommand\nofixcompositeaccent{%
2706 \let\oalign\pltx@sav@d@oalign
2707 \let\ltx@sh@ft\pltx@sav@d@ltx@sh@ft
2708 \let\@text@composite@x\pltx@sav@d@text@composite@x
2709 }

```

```

2710 \nofixcompositeaccent
2711 </plcore| platexrelease>
2712 <platexrelease>%\plEndIncludeInRelease

```

\@text@composite@x エミュレーション専用のコードです。

```

2713 <platexrelease>\plIncludeInRelease{2018/07/28}{\fixcompositeaccent}
2714 <platexrelease> {Fix for non-zero baselineshift}%
2715 <platexrelease>\nofixcompositeaccent % force LaTeX original (conditional default)
2716 <platexrelease>% other commands are actually defined for pLaTeX2e 2018-07-28
2717 <platexrelease>\plEndIncludeInRelease
2718 <platexrelease>\plIncludeInRelease{2016/07/01}{\fixcompositeaccent}
2719 <platexrelease> {Fix for non-zero baselineshift}%
2720 <platexrelease>\nofixcompositeaccent % force LaTeX original (always)
2721 <platexrelease>\let\fixcompositeaccent\@undefined
2722 <platexrelease>\let\nofixcompositeaccent\@undefined
2723 <platexrelease>\let\pltx@saved@oalign\@undefined
2724 <platexrelease>\let\pltx@oalign\@undefined
2725 <platexrelease>\let\pltx@saved@ltx@sh@ft\@undefined
2726 <platexrelease>\let\pltx@ltx@sh@ft\@undefined
2727 <platexrelease>\let\pltx@saved@text@composite@x\@undefined
2728 <platexrelease>\let\pltx@text@composite@x\@undefined
2729 <platexrelease>\plEndIncludeInRelease
2730 <platexrelease>\plIncludeInRelease{2016/04/17}{\fixcompositeaccent}
2731 <platexrelease> {Fix for non-zero baselineshift}%
2732 <platexrelease>\fixcompositeaccent % force pLaTeX definition (always)
2733 <platexrelease>\let\oalign\pltx@saved@oalign % no fix at that time
2734 <platexrelease>\let\ltx@sh@ft\pltx@saved@ltx@sh@ft % no fix at that time
2735 <platexrelease>\let\fixcompositeaccent\@undefined
2736 <platexrelease>\let\nofixcompositeaccent\@undefined
2737 <platexrelease>\let\pltx@saved@oalign\@undefined
2738 <platexrelease>\let\pltx@oalign\@undefined
2739 <platexrelease>\let\pltx@saved@ltx@sh@ft\@undefined
2740 <platexrelease>\let\pltx@ltx@sh@ft\@undefined
2741 <platexrelease>\let\pltx@saved@text@composite@x\@undefined
2742 <platexrelease>\let\pltx@text@composite@x\@undefined
2743 <platexrelease>\plEndIncludeInRelease
2744 <platexrelease>\plIncludeInRelease{0000/00/00}{\fixcompositeaccent}
2745 <platexrelease> {Fix for non-zero baselineshift}%
2746 <platexrelease>\nofixcompositeaccent % force LaTeX original (always)
2747 <platexrelease>\let\fixcompositeaccent\@undefined
2748 <platexrelease>\let\nofixcompositeaccent\@undefined
2749 <platexrelease>\let\pltx@saved@oalign\@undefined
2750 <platexrelease>\let\pltx@oalign\@undefined
2751 <platexrelease>\let\pltx@saved@ltx@sh@ft\@undefined
2752 <platexrelease>\let\pltx@ltx@sh@ft\@undefined
2753 <platexrelease>\let\pltx@saved@text@composite@x\@undefined
2754 <platexrelease>\let\pltx@text@composite@x\@undefined
2755 <platexrelease>\plEndIncludeInRelease

```

7.6 イタリック補正と \xkanjiskip

\check@nocorr@ 「あ \texttt{abc}い」としたとき、書体の変更を指定された欧文の左側に和欧文間スペースが入らないのを修正します。

コミュニティ版の修正：pTeX のバージョン p3.1.11 以前は、イタリック補正（以下 \/ と記す）と \xkanjiskip の挿入が衝突²し

1. 「欧文文字 → \/」の場合には \/ を無視する（つまり後に \xkanjiskip 挿入可能）
2. 「和文文字 → \/」の場合にはこの後に \xkanjiskip は挿入できない

という挙動になっていました。p3.2（2010 年）の修正で

- \xkanjiskip 挿入時にはいかなる場合も \/ を無視する

という挙動に変更されました。pLaTeX カーネルの \check@nocorr@ の修正は、p3.1.11 以前の 2. への対処でしたが、これは「\text...{} の左への \/ 挿入」を無効化しているので、\textit{f\textup{a}} で本来入るべきイタリック補正が入りませんでした。p3.2 以降では pTeX の \xkanjiskip 対策が不要になっていますので、コミュニティ版では削除しました。

```

2756 <platexrelease>\plIncludeInRelease{2017/10/28}{\check@nocorr@}
2757 <platexrelease>          {Italic correction before \textt...}%
2758 <platexrelease>\def \check@nocorr@ #1#2\nocorr#3\@nil {%
2759 <platexrelease>  \let \check@ic1 \maybe@ic
2760 <platexrelease>  \def \check@icr {\ifvmode \else \aftergroup \maybe@ic \fi}%
2761 <platexrelease>  \def \reserved@a {\nocorr}%
2762 <platexrelease>  \def \reserved@b {#1}%
2763 <platexrelease>  \def \reserved@c {#3}%
2764 <platexrelease>  \ifx \reserved@a \reserved@b
2765 <platexrelease>    \ifx \reserved@c \@empty
2766 <platexrelease>      \let \check@ic1 \@empty
2767 <platexrelease>    \else
2768 <platexrelease>      \let \check@ic1 \@empty
2769 <platexrelease>      \let \check@icr \@empty
2770 <platexrelease>    \fi
2771 <platexrelease>  \else
2772 <platexrelease>    \ifx \reserved@c \@empty
2773 <platexrelease>      \else
2774 <platexrelease>        \let \check@icr \@empty
2775 <platexrelease>      \fi
2776 <platexrelease>    \fi
2777 <platexrelease>}
2778 <platexrelease>\plEndIncludeInRelease
2779 <platexrelease>\plIncludeInRelease{0000/00/00}{\check@nocorr@}
2780 <platexrelease>          {ASCII Corporation original}%
2781 <platexrelease>\def \check@nocorr@ #1#2\nocorr#3\@nil {%
2782 <platexrelease>  \let \check@ic1 \relax % changed from \maybe@ic
2783 <platexrelease>  \def \check@icr {\ifvmode \else \aftergroup \maybe@ic \fi}%
2784 <platexrelease>  \def \reserved@a {\nocorr}%
2785 <platexrelease>  \def \reserved@b {#1}%
2786 <platexrelease>  \def \reserved@c {#3}%
2787 <platexrelease>  \ifx \reserved@a \reserved@b

```

²和文のイタリック補正用 kern が、通常の explicit な (\kern による) kern と同じ扱いを受けていたため。

```

2788 <platexrelease> \ifx \reserved@c \@empty
2789 <platexrelease> \let \check@ic1 \@empty
2790 <platexrelease> \else
2791 <platexrelease> \let \check@ic1 \@empty
2792 <platexrelease> \let \check@icr \@empty
2793 <platexrelease> \fi
2794 <platexrelease> \else
2795 <platexrelease> \ifx \reserved@c \@empty
2796 <platexrelease> \else
2797 <platexrelease> \let \check@icr \@empty
2798 <platexrelease> \fi
2799 <platexrelease> \fi
2800 <platexrelease> }
2801 <platexrelease> \plEndIncludeInRelease

```

\< 最後に、\inhibitglue の簡略形を定義します。このコマンドは、和文フォントのメトリック情報から、自動的に挿入されるグルーの挿入を禁止します。

2014 年の pT_EX の \inhibitglue のバグ修正に伴い、\inhibitglue が垂直モードでは効かなくなりました。L^AT_EX では垂直モードと水平モードの区別が隠されていますので、pL^AT_EX の追加命令である \< は段落頭でも効くように修正します。

\DeclareRobustCommand を使うと \protect の影響で前方の文字に対する \inhibitglue が効かなくなるので、e-T_EX の \protected が必要です。

```

2802 <platexrelease> \plIncludeInRelease{2017/10/28}{\<}
2803 <platexrelease> \inhibitglue in vertical mode}%
2804 <plcore | platexrelease>
2805 \ifx \protected \@undefined
2806 \def \< {\inhibitglue}
2807 \else
2808 \protected \def \< {\ifvmode \leavevmode \fi \inhibitglue}
2809 \fi
2810 </plcore | platexrelease>
2811 <platexrelease> \plEndIncludeInRelease
2812 <platexrelease> \plIncludeInRelease{0000/00/00}{\<}
2813 <platexrelease> \ASCII Corporation original}%
2814 <platexrelease> \def \< {\inhibitglue}
2815 <platexrelease> \plEndIncludeInRelease

```

7.7 デフォルト設定ファイルの読み込み

デフォルト設定ファイル pldefs.ltx は、もともと plcore.ltx の途中で読み込んでいましたが、2018 年以降の新しいコミュニティ版 pL^AT_EX では platex.ltx から読み込むことにしました。実際の中身については、第 8 節を参照してください。

8 デフォルト設定ファイル

ここでは、フォーマットファイルに読み込まれるデフォルト値を設定しています。この節での内容は pldefs.ltx に出力されます。このファイルの内容を plcore.ltx

に含めてもよいのですが、デフォルトの設定を参照しやすいように、別ファイルに
してあります。

プリロードサイズは、DOCSTRIP プログラムのオプションで変更することができます。
これ以外の変更したい場合は、pldefs.ltx を直接、修正するのではなく、この
ファイルを pldefs.cfg という名前でコピーをして、そのファイルに対して修正を
加えるようにしてください。

```
2816 (*pldefs)
2817 \ProvidesFile{pldefs.ltx}
2818 [2021/01/10 v1.7k pLaTeX Kernel (Default settings)]
2819 \</pldefs>
```

8.1 テキストフォント

テキストフォントのための属性やエラー書体などの宣言です。pLaTeX のデフォルト
の横組エンコードは JY1、縦組エンコードは JT1 とします。縦横エンコード共通：

```
2820 (*pldefs)
2821 \DeclareKanjiEncodingDefaults{}{}
2822 \DeclareErrorKanjiFont{JY1}{mc}{m}{n}{10}
2823 \kanjifamily{mc}
2824 \def\k@series{m} % \kanjiseriess{m}
2825 \def\k@shape{n} % \kanjishape{n}
2826 \fontsize{10}{10}
```

横組エンコード：

```
2827 \DeclareYokoKanjiEncoding{JY1}{}{}
2828 \DeclareKanjiSubstitution{JY1}{mc}{m}{n}
```

縦組エンコード：

```
2829 \DeclareTateKanjiEncoding{JT1}{}{}
2830 \DeclareKanjiSubstitution{JT1}{mc}{m}{n}
```

縦横のエンコーディングのセット化：

```
2831 \KanjiEncodingPair{JY1}{JT1}
```

フォント属性のデフォルト値:LaTeX 2_ε 2019-10-01 までは \shapedefault は \updefault
でしたが、LaTeX 2_ε 2020-02-02 で \updefault が “n” から “up” へと修正されたこ
とに伴い、\shapedefault は明示的に “n” に設定されました。

```
2832 \newcommand\mcdefault{mc}
2833 \newcommand\gtdefault{gt}
2834 \newcommand\kanjiencodingdefault{JY1}
2835 \newcommand\kanjifamilydefault{\mcdefault}
2836 \newcommand\kanjiseriessdefault{\mddefault}
2837 \newcommand\kanjishapedefault{n}% formerly \updefault
```

和文エンコードの指定：

```
2838 \kanjiencoding{JY1}
```

フォント定義：これらの具体的な内容は第 9 節を参照してください。

```
2839 \input{jy1mc.fd}
2840 \input{jy1gt.fd}
2841 \input{jt1mc.fd}
2842 \input{jt1gt.fd}
```

フォントを有効にします。

```
2843 \fontencoding{JT1}\selectfont
2844 \fontencoding{JY1}\selectfont
```

8.2 プリロードフォント

あらかじめフォーマットファイルにロードされるフォントの宣言です。DOCSTRIP プログラムのオプションでロードされるフォントのサイズを変更することができます。

plfmt.ins では xpt を指定しています。

```
2845 ⟨*xpt⟩
2846 \DeclarePreloadSizes{JY1}{mc}{m}{n}{5,7,10,12}
2847 \DeclarePreloadSizes{JY1}{gt}{m}{n}{5,7,10,12}
2848 \DeclarePreloadSizes{JT1}{mc}{m}{n}{5,7,10,12}
2849 \DeclarePreloadSizes{JT1}{gt}{m}{n}{5,7,10,12}
2850 ⟨/xpt⟩
2851 ⟨*xipt⟩
2852 \DeclarePreloadSizes{JY1}{mc}{m}{n}{5,7,10.95,12}
2853 \DeclarePreloadSizes{JY1}{gt}{m}{n}{5,7,10.95,12}
2854 \DeclarePreloadSizes{JT1}{mc}{m}{n}{5,7,10.95,12}
2855 \DeclarePreloadSizes{JT1}{gt}{m}{n}{5,7,10.95,12}
2856 ⟨/xipt⟩
2857 ⟨*xiipt⟩
2858 \DeclarePreloadSizes{JY1}{mc}{m}{n}{7,9,12,14.4}
2859 \DeclarePreloadSizes{JY1}{gt}{m}{n}{7,9,12,14.4}
2860 \DeclarePreloadSizes{JT1}{mc}{m}{n}{7,9,12,14.4}
2861 \DeclarePreloadSizes{JT1}{gt}{m}{n}{7,9,12,14.4}
2862 ⟨/xiipt⟩
2863 ⟨*ori⟩
2864 \DeclarePreloadSizes{JY1}{mc}{m}{n}
2865         {5,6,7,8,9,10,10.95,12,14.4,17.28,20.74,24.88}
2866 \DeclarePreloadSizes{JY1}{gt}{m}{n}
2867         {5,6,7,8,9,10,10.95,12,14.4,17.28,20.74,24.88}
2868 \DeclarePreloadSizes{JT1}{mc}{m}{n}
2869         {5,6,7,8,9,10,10.95,12,14.4,17.28,20.74,24.88}
2870 \DeclarePreloadSizes{JT1}{gt}{m}{n}
2871         {5,6,7,8,9,10,10.95,12,14.4,17.28,20.74,24.88}
2872 ⟨/ori⟩
```

8.3 組版パラメータ

禁則パラメータや文字間へ挿入するスペースの設定などです。実際の各文字への禁則パラメータおよびスペースの挿入の許可設定などは、kinsoku.tex で行なっています。具体的な設定については、kinsoku.dtx を参照してください。

```
2873 \InputIfFileExists{kinsoku.tex}%
2874 {\message{Loading kinsoku patterns for japanese.}}
2875 {\errhelp{The configuration for kinsoku is incorrectly installed.^^J%
2876         If you don't understand this error message you need
2877         to seek^^Jexpert advice.}%
2878 \errmessage{OOPS! I can't find any kinsoku patterns for japanese^^J%
2879         \space Think of getting some or the
2880         latex2e setup will never succeed}\@@end}
```

組版パラメータの設定をします。`\kanjiskip` は、漢字と漢字の間に挿入されるグルーです。`\noautospacing` で、挿入を中止することができます。デフォルトは `\autospacing` です。

```
2881 \kanjiskip=0pt plus .4pt minus .5pt
2882 \autospacing
```

`\xkanjiskip` は、和欧文間に自動的に挿入されるグルーです。`\noautoxspacing` で、挿入を中止することができます。デフォルトは `\autoxspacing` です。

```
2883 \xkanjiskip=.25zw plus1pt minus1pt
2884 \autoxspacing
```

`\jcharwidowpenalty` は、パラグラフに対する禁則です。パラグラフの最後の行が 1 文字だけにならないように調整するために使われます。

```
2885 \jcharwidowpenalty=500
```

ここまでが、`pldefs.ltx` の内容です。

```
2886 \end{pldefs}
```

9 フォント定義ファイル

ここでは、フォント定義ファイルの設定をしています。フォント定義ファイルは、 \LaTeX のフォント属性を \TeX フォントに置き換えるためのファイルです。記述方法についての詳細は、`fntguide.tex` を参照してください。

欧文書体の設定については、`cmfonts.fdd` や `slides.fdd` などを参照してください。`skfonts.fdd` には、写研代用書体を使うためのパッケージとフォント定義が記述されています。

```
2887 \JYlmc\ProvidesFile{jy1mc.fd}
2888 \JYlgt\ProvidesFile{jy1gt.fd}
2889 \JTlmc\ProvidesFile{jt1mc.fd}
2890 \JTlgt\ProvidesFile{jt1gt.fd}
2891 \JYlmc,JYlgt,JTlmc,JTlgt) [2018/07/03 v1.6q KANJI font defines]
```

横組用、縦組用ともに、明朝体のシリーズ `bx` がゴシック体となるように宣言しています。また、シリーズ `b` は同じ書体の `bx` と等価になるように宣言します。

\pLaTeX では従属書体に OT1 エンコーディングを指定しています。また、要求サイズ（指定されたフォントサイズ）が 10pt のとき、全角幅の実寸が 9.62216pt となるようにしますので、和文スケール値（ $1\text{zw} \div \text{要求サイズ}$ ）は $9.62216\text{pt}/10\text{pt} = 0.962216$ です。min10 系のメトリックは全角幅が 9.62216pt でデザインされているので、これを 1 倍で読み込みます。

```
2892 \*JYlmc)
2893 \DeclareKanjiFamily{JY1}{mc}{}
2894 \DeclareRelationFont{JY1}{mc}{m}{OT1}{cmr}{m}{}
2895 \DeclareRelationFont{JY1}{mc}{bx}{OT1}{cmr}{bx}{}
2896 \DeclareFontShape{JY1}{mc}{m}{n}{<5> <6> <7> <8> <9> <10> sgen*min
2897     <10.95><12><14.4><17.28><20.74><24.88> min10
2898     <-> min10
2899     }{}
```

```

2900 \DeclareFontShape{JY1}{mc}{bx}{n}{<->ssub*gt/m/n}{ }
2901 \DeclareFontShape{JY1}{mc}{b}{n}{<->ssub*mc/bx/n}{ }
2902 </JY1mc>
2903 <*JT1mc>
2904 \DeclareKanjiFamily{JT1}{mc}{ }
2905 \DeclareRelationFont{JT1}{mc}{m}{ }{OT1}{cmr}{m}{ }
2906 \DeclareRelationFont{JT1}{mc}{bx}{ }{OT1}{cmr}{bx}{ }
2907 \DeclareFontShape{JT1}{mc}{m}{n}{<5> <6> <7> <8> <9> <10> sgen*tmin
2908 <10.95><12><14.4><17.28><20.74><24.88> tmin10
2909 <-> tmin10
2910 }{ }
2911 \DeclareFontShape{JT1}{mc}{bx}{n}{<->ssub*gt/m/n}{ }
2912 \DeclareFontShape{JT1}{mc}{b}{n}{<->ssub*mc/bx/n}{ }
2913 </JT1mc>
2914 <*JY1gt>
2915 \DeclareKanjiFamily{JY1}{gt}{ }
2916 \DeclareRelationFont{JY1}{gt}{m}{ }{OT1}{cmr}{bx}{ }
2917 \DeclareFontShape{JY1}{gt}{m}{n}{<5> <6> <7> <8> <9> <10> sgen*goth
2918 <10.95><12><14.4><17.28><20.74><24.88> goth10
2919 <-> goth10
2920 }{ }
2921 \DeclareFontShape{JY1}{gt}{bx}{n}{<->ssub*gt/m/n}{ }
2922 \DeclareFontShape{JY1}{gt}{b}{n}{<->ssub*gt/bx/n}{ }
2923 </JY1gt>
2924 <*JT1gt>
2925 \DeclareKanjiFamily{JT1}{gt}{ }
2926 \DeclareRelationFont{JT1}{gt}{m}{ }{OT1}{cmr}{bx}{ }
2927 \DeclareFontShape{JT1}{gt}{m}{n}{<5> <6> <7> <8> <9> <10> sgen*tgoth
2928 <10.95><12><14.4><17.28><20.74><24.88> tgoth10
2929 <-> tgoth10
2930 }{ }
2931 \DeclareFontShape{JT1}{gt}{bx}{n}{<->ssub*gt/m/n}{ }
2932 \DeclareFontShape{JT1}{gt}{b}{n}{<->ssub*gt/bx/n}{ }
2933 </JT1gt>

```


File d

plcore.dtx

10 概要

このファイルでは、つぎの機能の拡張や修正を行っています。詳細は、それぞれの項目の説明を参照してください。

- プリアンブルコマンド
- 改ページ
- 改行
- オブジェクトの出力順序
- トンボ
- 出力ルーチン
- 脚注マクロ
- 相互参照
- 疑似タイプ入力
- tabbing 環境
- 用語集の出力
- 時分を示すカウンタ

11 コード

このファイルの内容は、pL^AT_EX 2_ε のコア部分です。

1 (*plcore)

11.1 プリアンブルコマンド

文書ファイルが必要とするフォーマットファイルの指定をするコマンドを拡張し、pL^AT_EX 2_ε フォーマットファイルも認識するようにします。

```
\NeedsTeXFormat \NeedsTeXFormats に “pLaTeX2e” を指定すると、“LaTeX2e” フォーマットを必要
\@needsPformat とする英語版のクラスファイルやパッケージファイルなどが使えなくなってしまう
\@needsPf@rmat ために再定義します。このコマンドは ltclass.dtx で定義されています。
2 \def\NeedsTeXFormat#1{%
3   \def\reserved@a{#1}%
```

```

4 \ifx\reserved@a\pfmtname
5 \expandafter\@needsPformat
6 \else
7 \ifx\reserved@a\fmtname
8 \expandafter\expandafter\expandafter\@needsformat
9 \else
10 \@latex@error{This file needs format '\reserved@a'%
11 \MessageBreak but this is '\pfmtname'}{%
12 The current input file will not be processed
13 further,\MessageBreak
14 because it was written for some other flavor of
15 TeX.\MessageBreak\@ehd}%
16 \endinput
17 \fi
18 \fi}
19 %
20 \def\@needsPformat{\@ifnextchar[\@needsPf@rmat{}}
21 %
22 \def\@needsPf@rmat[#1]{%
23 \ifl@t@r\pfmtversion{#1}{}%
24 {\@latex@warning@no@line
25 {You have requested release '#1' of pLaTeX,\MessageBreak
26 but only release '\pfmtversion' is available}}}%
27 %
28 \@onlypreamble\@needsPformat
29 \@onlypreamble\@needsPf@rmat

```

`\documentstyle` `\documentclass` の代わりに `\documentstyle` が使われると、 $\mathrm{\LaTeX}$ 2.09 互換モードに入ります。このとき、オリジナルの $\mathrm{\LaTeX}$ では `latex209.def` を読み込みますが、 $\mathrm{p\LaTeX}$ 2_ε では `pl209.def` を読み込みます。このコマンドは `ltxclass.dtx` で定義されています。

```

30 \def\documentstyle{%
31 \makeatletter\input{pl209.def}\makeatother
32 \documentclass}
33 \</plcore>

```

11.2 直前の JFM 由来スペースの削除【コミュニティ版独自】

現状の pTeX (TeX Live 2017 時点) では、`\inhibitglue` プリミティブは「JFM 由来のスペース (グルー・カーン) 挿入ルーチンを抑制する」働きをします。しかし、既に挿入されてしまった JFM グルーやカーンを削除することはできません。

`\removejfmglue` そこで、「最後のノードが JFM グルーであった場合にそれを削除する」というユーザ向け命令を定義します。この機能には $\mathrm{e-pTeX}$ 180226 以降の `\lastnodesubtype` プリミティブが必要です。この命令はあくまで「`\removejfmglue` の展開時点で既に pTeX によって挿入完了している JFM グルー」だけを削除し、「これから挿入されようとする JFM グルー」は抑制しません。例えば

始) `\removejfmglue` 中) `\relax\removejfmglue` 終

という入力からは

始) 中) 終

が得られます (最初の `\removejfmglue` は結果的に何もしていません)。

```
34 <latexrelease>\plIncludeInRelease{2018/03/09}%
35 <latexrelease>          {\removejfmglue}{Macro added}%
36 <*plcore | latexrelease>

37 \ifx\lastnodesubtype\@undefined
38   \let\removejfmglue\@undefined
39 \else
40   \setbox0\hbox{%
41     \ifnum\jis"2121="3000 %% upTeX check
42       \jfont\tenmin=upjisr-h at 9.62216pt
43     \else
44       \jfont\tenmin=min10
45     \fi\tenmin
46     \char\jis"214B\null\setbox0\lastbox
47     \global\chardef\pltx@gluetype\lastnodetype
48     \global\chardef\pltx@jfmgluesubtype\lastnodesubtype
49   }
50   \setbox0=\box\voidb@x
51   \protected\def\removejfmglue{%
52     \ifnum\lastnodetype=\pltx@gluetype\relax
53       \ifnum\lastnodesubtype=\pltx@jfmgluesubtype\relax
54         \unskip
55       \fi
56     \fi}
57 \fi
58 </plcore | latexrelease>
59 <latexrelease>\plEndIncludeInRelease
60 <latexrelease>\plIncludeInRelease{0000/00/00}%
61 <latexrelease>          {\removejfmglue}{Macro added}%
62 <latexrelease>\let\removejfmglue\@undefined
63 <latexrelease>\plEndIncludeInRelease
```

11.3 改ページ

縦組のとき、改ページ後の内容が偶数ページ (右ページ) からはじまるようにします。横組のときには、奇数ページ (右ページ) からはじまります。

`\cleardoublepage` このコマンドによって出力される、白ページのページスタイルを *empty* にし、ヘッダとフッタが入らないようにしています。ltoutput.dtx の定義を、縦組、横組に合わせて、定義しなおしたものです。

```
64 <*plcore>
65 \def\cleardoublepage{\clearpage\if@twoside
66   \ifodd\c@page
67     \iftdir
68       \hbox{}\thispagestyle{empty}\newpage
69     \if@twocolumn\hbox{}\newpage\fi
70   \fi
71 \else
72   \ifdir
```

```

73      \hbox{}\thispagestyle{empty}\newpage
74      \if@twocolumn\hbox{}\newpage\fi
75      \fi
76      \fi\fi}

```

11.4 改行

`\@gnewline` 日本語 \TeX の行頭禁則処理は、禁則対象文字の直前に、`\prebreakpenalty` で指定されたペナルティの値を挿入することで行なっています。ところが、改行コマンドは負のペナルティの値を挿入することで改行を行ないます。そのために、禁則ペナルティの値が 10000 の文字の直後では、ペナルティの値が相殺され、改行することができません。

```

あいうえお \\
!かきくけこ

```

したがって、`\newline` マクロに `\mbox{}` を入れることによって、`\newline` マクロのペナルティ-10000 と行頭文字のペナルティ 10000 が加算されないようにします。`\\` は `\newline` マクロを呼び出しています。

なお、`\newline` マクロは `ltspace.dtx` で定義されています。

$\mathrm{L}^{\mathrm{A}}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ <1996/12/01>で改行マクロが変更され、`\\` が `\newline` を呼び出さなくなったため、変更された改行マクロに対応しました。`\null` の挿入位置は同じです。`ltspace.dtx` の定義を上記に合わせて、定義しなおしました。

日本語 \TeX 開発コミュニティによる補足：アスキーによる $\mathrm{p}^{\mathrm{A}}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ では、行頭禁則文字の直前で `\\` による強制改行を行えるようにするという目的で `\null` を `\@gnewline` マクロ内に挿入していました。しかし、これでは `\\par` と書いた場合に Underfull 警告が出なくなっています (`tests/newline_par.tex` を `latex` と `platex` で処理してみてください)。

もし `\null` の代わりに `\hskip\z@` を挿入すれば、 $\mathrm{L}^{\mathrm{A}}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ と同様に Underfull 警告を出すことができます。ただし、`\null` を挿入した場合と異なり、強制改行後の行頭に JFM グルーが入らなくなります。これはむしろ、奥村さんの `jsclasses` で行頭を天ツキに直しているのと同じですが、 $\mathrm{p}^{\mathrm{A}}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ としては挙動が変化してしまいますので、現時点では `\null` \rightarrow `\hskip\z@` への変更を見送っています。

```

77 \def\@gnewline #1{%
78   \ifvmode
79     \@nolnerr
80   \else
81     \unskip \reserved@e {\reserved@f#1}\nobreak \hfil \break \null
82     \ignorespaces
83   \fi}
84 \plcore

```

`\@no@lnbk` **日本語 \TeX 開発コミュニティによる追加：**さらに、`\\` だけでなく `\linebreak` についても同様の対処をします。 $\mathrm{L}^{\mathrm{A}}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ の定義のままではマクロによるペナルティ-10000 と行頭文字のペナルティ 10000 が加算されてしまうため、`\hskip\z@\relax` を入れ

ておきます。なお、`\linebreak`を発行して行分割が起きた場合、新しい行頭のJFMグルーは消えるという従来の`pLATEX`の挙動も維持しています。

前回の`\hskip\z@\relax`の追加では、`\nolinebreak`の場合に`\kanjiskip`や`\xkanjiskip`が入らない問題が起きてしまいました。そこで、`\penalty\z@\relax`に変更しました。これは、明示的な`\penalty`プリミティブ同士の合算は行われないうことを利用しています。

ところが、その変更によってそもそも`\nolinebreak`が効かない場合が生じたので、変更全体をいったんキャンセルして元に戻します。

```

85 <latexrelease>\plIncludeInRelease{2017/10/28}{\@no@lnbk}
86 <latexrelease>          {Break before prebreakpenalty (revert)}%
87 <latexrelease>\def\@no@lnbk #1[#2]{%
88 <latexrelease>  \ifvmode
89 <latexrelease>    \@nolnerr
90 <latexrelease>  \else
91 <latexrelease>    \@tempskipa\lastskip
92 <latexrelease>    \unskip
93 <latexrelease>    \penalty #1\@getpen{#2}%
94 <latexrelease>    \ifdim\@tempskipa>\z@
95 <latexrelease>      \hskip\@tempskipa
96 <latexrelease>      \ignorespaces
97 <latexrelease>    \fi
98 <latexrelease>  \fi}
99 <latexrelease>\plEndIncludeInRelease
100 <latexrelease>\plIncludeInRelease{2017/07/29}{\@no@lnbk}
101 <latexrelease>          {Break before prebreakpenalty (another)}%
102 <latexrelease>\def\@no@lnbk #1[#2]{%
103 <latexrelease>  \ifvmode
104 <latexrelease>    \@nolnerr
105 <latexrelease>  \else
106 <latexrelease>    \@tempskipa\lastskip
107 <latexrelease>    \unskip
108 <latexrelease>    \penalty #1\@getpen{#2}%
109 <latexrelease>    \penalty\z@\relax %% added (2017/08/25)
110 <latexrelease>    \ifdim\@tempskipa>\z@
111 <latexrelease>      \hskip\@tempskipa
112 <latexrelease>      \ignorespaces
113 <latexrelease>    \fi
114 <latexrelease>  \fi}
115 <latexrelease>\plEndIncludeInRelease
116 <latexrelease>\plIncludeInRelease{2017/05/05}{\@no@lnbk}
117 <latexrelease>          {Break before prebreakpenalty}%
118 <latexrelease>\def\@no@lnbk #1[#2]{%
119 <latexrelease>  \ifvmode
120 <latexrelease>    \@nolnerr
121 <latexrelease>  \else
122 <latexrelease>    \@tempskipa\lastskip
123 <latexrelease>    \unskip
124 <latexrelease>    \penalty #1\@getpen{#2}%
125 <latexrelease>    \hskip\z@\relax %% added (2017/05/03)
126 <latexrelease>    \ifdim\@tempskipa>\z@
127 <latexrelease>      \hskip\@tempskipa

```

```

128 <platexrelease> \ignorespaces
129 <platexrelease> \fi
130 <platexrelease> \fi}
131 <platexrelease>\plEndIncludeInRelease
132 <platexrelease>\plIncludeInRelease{0000/00/00}{\@no@lnbk}
133 <platexrelease> {LaTeX2e original}%
134 <platexrelease>\def\@no@lnbk #1[#2]{%
135 <platexrelease> \ifvmode
136 <platexrelease> \@nolnerr
137 <platexrelease> \else
138 <platexrelease> \@tempskipa\lastskip
139 <platexrelease> \unskip
140 <platexrelease> \penalty #1\getpen{#2}%
141 <platexrelease> \ifdim\@tempskipa>\z@
142 <platexrelease> \hskip\@tempskipa
143 <platexrelease> \ignorespaces
144 <platexrelease> \fi
145 <platexrelease> \fi}
146 <platexrelease>\plEndIncludeInRelease

```

なお、 \LaTeX 用の命令である $\backslash\backslash$ と $\backslash\text{linebreak}$ には上記のような禁則文字への対策を入っていますが、plain \TeX 互換のシンプルな命令である $\backslash\text{break}$ や $\backslash\text{nobreak}$ には、対策を行いません。

11.5 オブジェクトの出力順序

オリジナルの \LaTeX は、トップフロート、本文、脚注、ボトムフロートの順番で出力しますが、日本語組版では、トップフロート、本文、ボトムフロート、脚注という順番の方が一般的ですので、このような順番になるよう修正をします。

したがって、文書ファイルによっては \LaTeX の組版結果と異なる場合がありますので、注意をしてください。

2014 年に \LaTeX に `fltrace` パッケージが追加されましたので、その $\text{p}\text{\LaTeX}$ 版として `pfltrace` パッケージを追加します。この `pfltrace` パッケージは \LaTeX の `fltrace` パッケージに依存します。

```

147 <fltrace>
148 \NeedsTeXFormat{pLaTeX2e}
149 \ProvidesPackage{pfltrace}
150 [2016/05/20 v1.2e Standard pLaTeX package (float tracing)]
151 \RequirePackageWithOptions{fltrace}
152 </fltrace>

```

$\backslash\text{pltx@adjust@wd@outputbox}$ $\backslash\text{@outputpage}$ 内で実行されていた

縦組の際に $\backslash\text{@outputbox}$ の内容が空のボックスだけの場合に、 $\backslash\text{wd}\backslash\text{@outputbox}$ が 0pt になってしまい、結果としてフッタの位置がくるってしまっていた。0 の $\backslash\text{hskip}$ を発生させると $\backslash\text{wd}\backslash\text{@outputbox}$ の値が期待したものとなるので、縦組の場合はその方法で対処する。

ただし、0 の `\hskip` を発生させるとき、水平モードに入ってしまうと、たとえば `longtable` パッケージを使用して表組途中で改ページするとき `\par -> {\vskip}` の無限ループが起きてしまいます。そこで、`\vbox` の中で発生させます。

という処理を取り出したものです。

LaTeX 2_ε 2021-06-01 では段落開始時の “para/*” フックが実装されますが、それを一時的に無効化するために「プリミティブとしての」`\everypar` を `\pdfprimitive\everypar` として呼び出しています。

```
153 <*plcore | latexrelease>
154 \def\pltx@adjust@wd@outputbox{%
155   \ifdir\else\vbox{\pdfprimitive\everypar{}\hskip\z@\fi}
```

`\pltx@adjust@wd@outputbox@vtryfc` `\pltx@adjust@wd@outputbox` と同様の処理ですが、`\vtryfc` では `\vbox` の位置が異なります。

```
156 \def\pltx@adjust@wd@outputbox@vtryfc{%
157   \ifdir\else\pdfprimitive\everypar{}\hskip\z@\fi}
158 </plcore | latexrelease>
```

`\@makecol` このマクロが組み立てる部分の中心となります。 `ltoutput.dtx` で定義されているものです。

```
159 <latexrelease>\plIncludeInRelease{2017/04/08}{\@makecol}
160 <latexrelease>          {Take into account depth of footnote}%
161 <*plcore | latexrelease>
162 \gdef\@makecol{%
163   \setbox\@outputbox\box\@cclv%
164   \let\@elt\relax % added on LaTeX (ltoutput.dtx 2003/12/16 v1.2k)
165   \xdef\@freelist{\@freelist\@midlist}%
166   \global \let \@midlist \@empty
167   \@combinefloats
```

オリジナルの LaTeX は、トップフロート、本文、脚注、ボトムフロートの順番で出力します。一方 pLaTeX は、トップフロート、本文、ボトムフロート、脚注の順番で出力します。ところが、アスキー版のコードは順番を入れ替えるだけでなく、脚注のあるページの版面全体の垂直位置が（特に縦組で顕著に）ずれてしまっていました。これは補正量 `\dp\@outputbox` の取得を脚注挿入より前に行っていたためで、コミュニティ版 pLaTeX ではこの問題に対処してあります。結果的に、`fnpos` パッケージ (`yafoot`) の `\makeFNbottom` かつ `\makeFNbelow` な状態と完全に等価になりました。

```
168   \let\pltx@textbottom\@textbottom % save (pLaTeX 2017/02/25)
169   \ifvoid\footins\else % changed (pLaTeX 2017/02/25)
170     \setbox\@outputbox \vbox {%
171       \boxmaxdepth \@maxdepth
172       \unvbox \@outputbox
173       \@textbottom % inserted here (pLaTeX 2017/02/25)
174       \vskip \skip\footins
175       \color@begingroup
176       \normalcolor
```

```

177         \footnoterule
178         \unvbox \footins
179         \color@endgroup
180     }%
181     \let\@textbottom\relax % disable temporarily (pLaTeX 2017/02/25)
182 \fi
183 \ifvbox\@kludgeins
184     \@makespecialcolbox
185 \else
186     \setbox\@outputbox \vbox to\@colht {%
187         %\boxmaxdepth \@maxdepth % comment out on LaTeX 1997/12/01
188         \@texttop
189         \dimen@ \dp\@outputbox
190         \unvbox \@outputbox

```

次の行は以前は

```
\iftdir\vbox{\hskip\z@}\fi
```

でしたが、\pltx@adjust@wd@outputbox として切り出しました。

```

191     \pltx@adjust@wd@outputbox
192     \vskip -\dimen@
193     \@textbottom
194 }%
195 \fi
196 \let\@textbottom\pltx@textbottom % restore (pLaTeX 2017/02/25)
197 \global \maxdepth \@maxdepth
198 }
199 </plcore | platexrelease>
200 <platexrelease>\plEndIncludeInRelease
201 <platexrelease>\plIncludeInRelease{2016/09/03}{\@makecol}
202 <platexrelease>          {Avoid infinite loop}%
203 <platexrelease>\gdef\@makecol{%
204 <platexrelease>    \setbox\@outputbox\box\@cclv%
205 <platexrelease>    \xdef\@freelist{\@freelist\@midlist}%
206 <platexrelease>    \global \let \@midlist \@empty
207 <platexrelease>    \@combinefloats
208 <platexrelease>    \ifvbox\@kludgeins
209 <platexrelease>        \@makespecialcolbox
210 <platexrelease>    \else
211 <platexrelease>        \setbox\@outputbox \vbox to\@colht {%
212 <platexrelease>            %\boxmaxdepth \@maxdepth % comment out on LaTeX 1997/12/01
213 <platexrelease>            \@texttop
214 <platexrelease>            \dimen@ \dp\@outputbox
215 <platexrelease>            \unvbox \@outputbox
216 <platexrelease>            \iftdir\vbox{\hskip\z@}\fi
217 <platexrelease>            \vskip -\dimen@
218 <platexrelease>            \@textbottom
219 <platexrelease>        \ifvoid\footins\else % for pLaTeX
220 <platexrelease>            \vskip \skip\footins
221 <platexrelease>            \color@begingroup
222 <platexrelease>                \normalcolor
223 <platexrelease>                \footnoterule
224 <platexrelease>                \unvbox \footins
225 <platexrelease>            \color@endgroup

```



```

226 <platexrelease>      \fi
227 <platexrelease>      }%
228 <platexrelease>      \fi
229 <platexrelease>      \global \maxdepth \@maxdepth
230 <platexrelease>}
231 <platexrelease>\plEndIncludeInRelease
232 <platexrelease>\plIncludeInRelease{2016/04/17}{\@makecol}
233 <platexrelease>      {Adjust for \dp\@outputbox in tate mode}%
234 <platexrelease>\gdef\@makecol{%
235 <platexrelease>      \setbox\@outputbox\box\@cclv%
236 <platexrelease>      \xdef\@freelist{\@freelist\@midlist}%
237 <platexrelease>      \global \let \@midlist \@empty
238 <platexrelease>      \@combinefloats
239 <platexrelease>      \ifvbox\@kludgeins
240 <platexrelease>      \@makespecialcolbox
241 <platexrelease>      \else
242 <platexrelease>      \setbox\@outputbox \vbox to\@colht {%
243 <platexrelease>      %\boxmaxdepth \@maxdepth      % comment out on LaTeX 1997/12/01
244 <platexrelease>      \@texttop
245 <platexrelease>      \dimen@ \dp\@outputbox
246 <platexrelease>      \unvbox \@outputbox
247 <platexrelease>      \iftdir\hskip\z@\fi
248 <platexrelease>      \vskip -\dimen@
249 <platexrelease>      \@textbottom
250 <platexrelease>      \ifvoid\footins\else % for pLaTeX
251 <platexrelease>      \vskip \skip\footins
252 <platexrelease>      \color@begingroup
253 <platexrelease>      \normalcolor
254 <platexrelease>      \footnoterule
255 <platexrelease>      \unvbox \footins
256 <platexrelease>      \color@endgroup
257 <platexrelease>      \fi
258 <platexrelease>      }%
259 <platexrelease>      \fi
260 <platexrelease>      \global \maxdepth \@maxdepth
261 <platexrelease>}
262 <platexrelease>\plEndIncludeInRelease
263 <platexrelease>\plIncludeInRelease{0000/00/00}{\@makecol}
264 <platexrelease>      {ASCII Corporation original}%
265 <platexrelease>\gdef\@makecol{%
266 <platexrelease>      \setbox\@outputbox\box\@cclv%
267 <platexrelease>      \xdef\@freelist{\@freelist\@midlist}%
268 <platexrelease>      \global \let \@midlist \@empty
269 <platexrelease>      \@combinefloats
270 <platexrelease>      \ifvbox\@kludgeins
271 <platexrelease>      \@makespecialcolbox
272 <platexrelease>      \else
273 <platexrelease>      \setbox\@outputbox \vbox to\@colht {%
274 <platexrelease>      %\boxmaxdepth \@maxdepth      % comment out on LaTeX 1997/12/01
275 <platexrelease>      \@texttop
276 <platexrelease>      \dimen@ \dp\@outputbox
277 <platexrelease>      \unvbox \@outputbox
278 <platexrelease>      \iftdir\hskip\z@
279 <platexrelease>      \else\vskip -\dimen@\fi

```

```

280 <platexrelease>      \@textbottom
281 <platexrelease>      \ifvoid\footins\else % for pLaTeX
282 <platexrelease>      \vskip \skip\footins
283 <platexrelease>      \color@begingroup
284 <platexrelease>      \normalcolor
285 <platexrelease>      \footnoterule
286 <platexrelease>      \unvbox \footins
287 <platexrelease>      \color@endgroup
288 <platexrelease>      \fi
289 <platexrelease>      }%
290 <platexrelease>      \fi
291 <platexrelease>      \global \maxdepth \@maxdepth
292 <platexrelease>      }
293 <platexrelease>      \plEndIncludeInRelease

```

`\@makespecialcolbox` 本文（あるいはボトムフロート）と脚注の間に `\@textbottom` を入れたいので、`\@makespecialcolbox` コマンドも修正をします。やはり、`ltoutput.dtx` で定義されているものです。

このマクロは、`\enlargethispage` が使われたときに、`\@makecol` マクロから呼び出されます。

日本語 $T_E X$ 開発コミュニティによる補足 (2017/02/25): 2016/11/29 以前の $p\text{L}^A T_E X$ では、`\@makecol` はボトムフロートを挿入した後、すぐに `\@kludgeins` が空かどうか判定し

- 空の場合は、残りすべての処理を `\@makespecialcolbox` に任せる
- 空でない場合は、`\@makecol` 自身で残りすべての処理を行う

としていました。しかし 2017/04/08 以降の $p\text{L}^A T_E X$ では、`\@makecol` はボトムフロートと脚注を挿入してから `\@kludgeins` の判定に移るようにしています。したがって、新しい `\@makecol` から以下に記す `\@makespecialcolbox` が呼び出される場合は、`\ifvoid\footins` (二箇所) の判定は常に真となるはずです。要するに「つぎの部分が $p\text{L}^A T_E X$ 用の修正です。」という二箇所のコードは実質的に不要となりました。

しかし、だからといって消してしまうと、古い $p\text{L}^A T_E X$ の `\@makecol` をベースに作られた外部パッケージから `\@makespecialcolbox` が呼び出される場合に脚注が消滅するおそれがあります。このため、`\@makespecialcolbox` は従来のコードのまま維持してあります (害はありません)。

```

294 (*plcore | fltrace)
295 \gdef\@makespecialcolbox{%
296 (*trace)
297   \fl@trace{Kludgeins ht \the\ht\@kludgeins\space
298             dp \the\dp\@kludgeins\space
299             wd \the\wd\@kludgeins}%
300 \trace)
301   \setbox\@outputbox \vbox {%
302     \texttop
303     \dimen@ \dp\@outputbox

```

```

304     \unvbox\@outputbox
305     \vskip-\dimen@
306     }%
307     \@tempdima \@colht
308     \ifdim \wd\@kludgeins>\z@
309         \advance \@tempdima -\ht\@outputbox
310         \advance \@tempdima \pageshrink
311 (*trace)
312     \fl@trace {Natural ht of col: \the\ht\@outputbox}%
313     \fl@trace {\string \@colht: \the\@colht}%
314     \fl@trace {Pageshrink added: \the\pageshrink}%
315     \fl@trace {Hence, space added: \the\@tempdima}%
316 (/trace)
317     \setbox\@outputbox \vbox to \@colht {%
318 %         \boxmaxdepth \maxdepth
319         \unvbox\@outputbox
320         \vskip \@tempdima
321         \@textbottom

```

つぎの部分が pL^AT_EX 用の修正です。

```

322     \ifvoid\footins\else % for pLATEX
323         \vskip\skip\footins
324         \color@begingroup
325         \normalcolor
326         \footnoterule
327         \unvbox \footins
328         \color@endgroup
329     \fi
330     }%
331     \else
332         \advance \@tempdima -\ht\@kludgeins
333 (*trace)
334     \fl@trace {Natural ht of col: \the\ht\@outputbox}%
335     \fl@trace {\string \@colht: \the\@colht}%
336     \fl@trace {Extra size added: -\the \ht \@kludgeins}%
337     \fl@trace {Hence, height of inner box: \the\@tempdima}%
338     \fl@trace {Max? pageshrink available: \the\pageshrink}%
339 (/trace)
340     \setbox \@outputbox \vbox to \@colht {%
341         \vbox to \@tempdima {%
342             \unvbox\@outputbox
343             \@textbottom

```

つぎの部分が pL^AT_EX 用の修正です。脚注があれば、ここでそれを出力します。

```

344     \ifvoid\footins\else % for pLATEX
345         \vskip\skip\footins
346         \color@begingroup
347         \normalcolor
348         \footnoterule
349         \unvbox \footins
350         \color@endgroup
351     \fi
352     }\vss}%
353 \fi

```

```

354   {\setbox \@tempboxa \box \@kludgeins}%
355 (*trace)
356   \fl@trace {kludgeins box made void}%
357 </trace>
358 }
359 </plcore | fltrace>

```

\@reinserts このマクロは、\@specialoutput マクロから呼び出されます。ボックス footins が組み立てられたモードに合わせて縦モードか横モードで \unvbox をします。

```

360 (*plcore)
361 \def\@reinserts{%
362   \ifvoid\footins\else\insert\footins{%
363     \iftbox\footins\tate\else\yoko\fi
364     \unvbox\footins}\fi
365   \ifvbox\@kludgeins\insert\@kludgeins{\unvbox\@kludgeins}\fi
366 }
367 </plcore>

```

\@vtryfc L^AT_EX 2_ε 2017/01/01 以降では、例えば

```

\documentclass{tarticle}
\begin{document}
\begin{figure}
\end{figure}
\clearpage
a
\end{document}

```

のようにすると「空のフロート」だけの空白ページが発生します。このとき、縦組クラスではフッタが持ち上がってしまうので、対策します。(Issue #78)

なお、L^AT_EX 2_ε 2015/01/01–2016/03/31 patch level 3 では

```
! Output loop---100 consecutive dead cycles.
```

のエラーが出ていました。それより昔の版では空白ページは発生しません。

対策方法は、ltoutput.dtx で定義されている \@vtryfc に \ifdir\else\hskip\z@\fi の追加です (\@makecol と同様) が、別命令 \pltx@adjust@wd@outputbox@vtryfc として切り出しました。

```

368 <platexrelease>\plIncludeInRelease{2020/10/01}{\@vtryfc}
369 <platexrelease>                                {Empty float}%
370 (*plcore | platexrelease)
371 \def\@vtryfc #1{%
372   \global\setbox\@outputbox\vbox{\pltx@adjust@wd@outputbox@vtryfc}%
373   \let\@elt\@wtryfc
374   \@flsucceed
375   \global\setbox\@outputbox \vbox to\@colht{%
376     \vskip \@fptop
377     \vskip -\@fpsep
378     \unvbox \@outputbox
379     \vskip \@fpbot}%
380   \let\@elt\relax
381   \xdef #1{\@failedlist\@flfail}%

```

```

382 \xdef\@freelist{\@freelist\@flsucceed}}
383 </plcore| platexrelease>
384 <platexrelease>\plEndIncludeInRelease
385 <platexrelease>\plIncludeInRelease{0000/00/00}{\@vtryfc}
386 <platexrelease> {LaTeX2e original}%
387 <platexrelease>\def\@vtryfc #1{%
388 <platexrelease> \global\setbox\@outputbox\vbox{%
389 <platexrelease> \let\@elt\@wtryfc
390 <platexrelease> \@flsucceed
391 <platexrelease> \global\setbox\@outputbox \vbox to\@colht{%
392 <platexrelease> \vskip \@fptop
393 <platexrelease> \vskip -\@fpsep
394 <platexrelease> \unvbox \@outputbox
395 <platexrelease> \vskip \@fpbot}%
396 <platexrelease> \let\@elt\relax
397 <platexrelease> \xdef #1{\@failedlist\@flfail}%
398 <platexrelease> \xdef\@freelist{\@freelist\@flsucceed}}
399 <platexrelease>\plEndIncludeInRelease

```

11.6 トンボ

ここではトンボを出力するためのマクロを定義しています。

`\iftombow` `\iftombow` はトンボを出力するかどうか、`\iftombowdate` は DVI を作成した日付 `\iftombowdate` をトンボの脇に出力するかどうかを示すために用います。

```

400 <*plcore>
401 \newif\iftombow \tombowfalse
402 \newif\iftombowdate \tombowdatetrue

```

`\@tombowwidth` `\@tombowwidth` には、トンボ用罫線の太さを指定します。デフォルトは 0.1 ポイントです。この値を変更し、`\maketombowbox` コマンドを実行することにより、トンボの罫線太さを変更して出力することができます。通常の使い方では、トンボの罫線を変更する必要はありません。DVI をフィルムに面付け出力するとき、トンボをつけずに位置はそのままにする必要があるときに、この太さをゼロポイントにします。

```

403 \newdimen\@tombowwidth
404 \setlength{\@tombowwidth}{.1\p@}

```

`\@tombowbleed` `\@tombowbleed` は、bleed 幅を指定します。デフォルトは 3mm です。

```

405 </plcore>
406 <platexrelease>\plIncludeInRelease{2018/05/20}{\@tombowbleed}{Macro added}%
407 <*plcore| platexrelease>
408 \def\@tombowbleed{3mm}
409 </plcore| platexrelease>
410 <platexrelease>\plEndIncludeInRelease
411 <platexrelease>\plIncludeInRelease{0000/00/00}{\@tombowbleed}{Macro added}%
412 <platexrelease>\let\@tombowbleed\undefined
413 <platexrelease>\plEndIncludeInRelease
414 <*plcore>

```

`\@tombowcolor` `\@tombowcolor` は、トンボの色です。デフォルトは `\normalcolor` です。

```

415 </plcore>
416 <latexrelease>\plIncludeInRelease{2018/05/20}{\@tombowcolor}{Macro added}%
417 <*plcore | latexrelease>
418 \def\@tombowcolor{\normalcolor}
419 </plcore | latexrelease>
420 <latexrelease>\plEndIncludeInRelease
421 <latexrelease>\plIncludeInRelease{0000/00/00}{\@tombowcolor}{Macro added}%
422 <latexrelease>\let\@tombowcolor\undefined
423 <latexrelease>\plEndIncludeInRelease
424 <*plcore>

```

トンボ用の罫線を定義します。

\@TL \@TL と \@Tl はページ上部の左側、\@TC はページ上部の中央、\@TR と \@Tr はページ上部の左側のトンボとなるボックスです。

```

\@TC 425 \newbox\@TL\newbox\@Tl
\@TR 426 \newbox\@TC
427 \newbox\@TR\newbox\@Tr
\@Tr

```

\@BL \@BL と \@Bl はページ下部の左側、\@BC はページ下部の中央、\@BR と \@Br はページ下部の左側のトンボとなるボックスです。

```

\@BC 428 \newbox\@BL\newbox\@Bl
429 \newbox\@BC
\@BR 430 \newbox\@BR\newbox\@Br
\@Br

```

\@CL \@CL はページ左側の中央、\@CR はページ右側の中央のトンボとなるボックスです。

```

\@CR 431 \newbox\@CL
432 \newbox\@CR

```

\@bannertoken \@bannertoken トークンは、トンボの横に出力する文字列を入れます。デフォルト \@bannerfont では何も出力しません。 \@bannerfont フォントは、その文字列を出力するためのフォントです。9ポイントのタイプライタ体としています。

```

433 \font\@bannerfont=cmtt9
434 \newtoks\@bannertoken
435 \@bannertoken{}

```

\maketombowbox \maketombowbox コマンドは、トンボとなるボックスを作るために用います。このコマンドは、トンボとなるボックスを作るだけで、それらのボックスを出力するのではないことに注意してください。

```

436 </plcore>
437 <latexrelease>\plIncludeInRelease{2018/05/20}{\maketombowbox}
438 <latexrelease> {Use \@tombowbleed}%
439 <*plcore | latexrelease>
440 \def\maketombowbox{%
441   \setbox\@TL\hbox to\z@{\yoko\hss
442     \vrule width\dimexpr 10mm+\@tombowbleed\relax height\@tombowwidth depth\z@
443     \vrule height10mm width\@tombowwidth depth\z@
444     \iftombowdate
445       \raise4pt\hbox to\z@{\hskip5mm\@bannerfont\the\@bannertoken\hss}%

```

```

446 \fi}%
447 \setbox\@TL\hbox to\z@\{yoko\hss
448 \vrule width10mm height\@tombowwidth depth\z@
449 \vrule height\dimexpr 10mm+\@tombowbleed\relax width\@tombowwidth depth\z@}%
450 \setbox\@TC\hbox\{yoko
451 \vrule width10mm height\@tombowwidth depth\z@
452 \vrule height10mm width\@tombowwidth depth\z@
453 \vrule width10mm height\@tombowwidth depth\z@}%
454 \setbox\@TR\hbox to\z@\{yoko
455 \vrule height10mm width\@tombowwidth depth\z@
456 \vrule width\dimexpr 10mm+\@tombowbleed\relax height\@tombowwidth depth\z@\hss}%
457 \setbox\@Tr\hbox to\z@\{yoko
458 \vrule height\dimexpr 10mm+\@tombowbleed\relax width\@tombowwidth depth\z@
459 \vrule width10mm height\@tombowwidth depth\z@\hss}%
460 %
461 \setbox\@BL\hbox to\z@\{yoko\hss
462 \vrule width\dimexpr 10mm+\@tombowbleed\relax depth\@tombowwidth height\z@
463 \vrule depth10mm width\@tombowwidth height\z@}%
464 \setbox\@Bl\hbox to\z@\{yoko\hss
465 \vrule width10mm depth\@tombowwidth height\z@
466 \vrule depth\dimexpr 10mm+\@tombowbleed\relax width\@tombowwidth height\z@}%
467 \setbox\@BC\hbox\{yoko
468 \vrule width10mm depth\@tombowwidth height\z@
469 \vrule depth10mm width\@tombowwidth height\z@
470 \vrule width10mm depth\@tombowwidth height\z@}%
471 \setbox\@BR\hbox to\z@\{yoko
472 \vrule depth10mm width\@tombowwidth height\z@
473 \vrule width\dimexpr 10mm+\@tombowbleed\relax depth\@tombowwidth height\z@\hss}%
474 \setbox\@Br\hbox to\z@\{yoko
475 \vrule depth\dimexpr 10mm+\@tombowbleed\relax width\@tombowwidth height\z@
476 \vrule width10mm depth\@tombowwidth height\z@\hss}%
477 %
478 \setbox\@CL\hbox to\z@\{yoko\hss
479 \vrule width10mm height.5\@tombowwidth depth.5\@tombowwidth
480 \vrule height10mm depth10mm width\@tombowwidth}%
481 \setbox\@CR\hbox to\z@\{yoko
482 \vrule height10mm depth10mm width\@tombowwidth
483 \vrule height.5\@tombowwidth depth.5\@tombowwidth width10mm\hss}%
484 }
485 \plcore| platexrelease)
486 \plEndIncludeInRelease
487 \plIncludeInRelease{0000/00/00}{\maketombowbox}
488 \plcore{ASCII Corporation original}%
489 \def\maketombowbox{%
490 \plcore \setbox\@TL\hbox to\z@\{yoko\hss
491 \plcore \vrule width13mm height\@tombowwidth depth\z@
492 \plcore \vrule height10mm width\@tombowwidth depth\z@
493 \plcore \iftombowdate
494 \plcore \raise4pt\hbox to\z@\{hskip5mm\@bannerfont\the\@bannertoken\hss}%
495 \plcore \fi}%
496 \plcore \setbox\@TL\hbox to\z@\{yoko\hss
497 \plcore \vrule width10mm height\@tombowwidth depth\z@
498 \plcore \vrule height13mm width\@tombowwidth depth\z@}%
499 \plcore \setbox\@TC\hbox\{yoko

```

```

500 <latexrelease> \vrule width10mm height\@tombowwidth depth\z@
501 <latexrelease> \vrule height10mm width\@tombowwidth depth\z@
502 <latexrelease> \vrule width10mm height\@tombowwidth depth\z@}%
503 <latexrelease> \setbox\@TR\hbox to\z@{\yoko
504 <latexrelease> \vrule height10mm width\@tombowwidth depth\z@
505 <latexrelease> \vrule width13mm height\@tombowwidth depth\z@\hss}%
506 <latexrelease> \setbox\@Tr\hbox to\z@{\yoko
507 <latexrelease> \vrule height13mm width\@tombowwidth depth\z@
508 <latexrelease> \vrule width10mm height\@tombowwidth depth\z@\hss}%
509 <latexrelease> \setbox\@BL\hbox to\z@{\yoko\hss
510 <latexrelease> \vrule width13mm depth\@tombowwidth height\z@
511 <latexrelease> \vrule depth10mm width\@tombowwidth height\z@}%
512 <latexrelease> \setbox\@Bl\hbox to\z@{\yoko\hss
513 <latexrelease> \vrule width10mm depth\@tombowwidth height\z@
514 <latexrelease> \vrule depth13mm width\@tombowwidth height\z@}%
515 <latexrelease> \setbox\@BC\hbox{\yoko
516 <latexrelease> \vrule width10mm depth\@tombowwidth height\z@
517 <latexrelease> \vrule depth10mm width\@tombowwidth height\z@
518 <latexrelease> \vrule width10mm depth\@tombowwidth height\z@}%
519 <latexrelease> \setbox\@BR\hbox to\z@{\yoko
520 <latexrelease> \vrule depth10mm width\@tombowwidth height\z@
521 <latexrelease> \vrule width13mm depth\@tombowwidth height\z@\hss}%
522 <latexrelease> \setbox\@Br\hbox to\z@{\yoko
523 <latexrelease> \vrule depth13mm width\@tombowwidth height\z@
524 <latexrelease> \vrule width10mm depth\@tombowwidth height\z@\hss}%
525 <latexrelease> \setbox\@CL\hbox to\z@{\yoko\hss
526 <latexrelease> \vrule width10mm height.5\@tombowwidth depth.5\@tombowwidth
527 <latexrelease> \vrule height10mm depth10mm width\@tombowwidth}%
528 <latexrelease> \setbox\@CR\hbox to\z@{\yoko
529 <latexrelease> \vrule height10mm depth10mm width\@tombowwidth
530 <latexrelease> \vrule height.5\@tombowwidth depth.5\@tombowwidth width10mm\hss}%
531 <latexrelease>}
532 <latexrelease>\plEndIncludeInRelease
533 <*plcore>

```

\@outputtombow \@outputtombow コマンドは、トンボを出力するのに用います。コミュニティ版では、「色付きテキストの途中で改ページが起きた場合に、トンボにも色が付いてしまう」という現象を防ぎ、さらにトンボの色を簡単に変えられるよう、\@tombowcolor というマクロに切り出しています。

```

534 </plcore>
535 <latexrelease>\plIncludeInRelease{2018/05/20}{\@outputtombow}
536 <latexrelease> {Use \@tombowcolor and \@tombowbleed}%
537 <*plcore | latexrelease>
538 \def\@outputtombow{%
539 \iftombow
540 \vbox to\z@{\kern-\dimexpr 10mm+\@tombowbleed\relax\relax
541 \boxmaxdepth\maxdimen
542 \moveleft\@tombowbleed \vbox to\@paperheight{%
543 \color@begingroup
544 \@tombowcolor
545 \hbox to\@paperwidth{\hskip\@tombowbleed\relax
546 \copy\@TL\hfill\copy\@TC\hfill\copy\@TR\hskip\@tombowbleed}%
547 \kern-10mm

```



```

548 \hbox to\@@paperwidth{\copy\@Tl\hfill\copy\@Tr}%
549 \vfill
550 \hbox to\@@paperwidth{\copy\@CL\hfill\copy\@CR}%
551 \vfill
552 \hbox to\@@paperwidth{\copy\@Bl\hfill\copy\@Br}%
553 \kern-10mm
554 \hbox to\@@paperwidth{\hskip\@tombowbleed\relax
555 \copy\@BL\hfill\copy\@BC\hfill\copy\@BR\hskip\@tombowbleed}%
556 \color@endgroup
557 }\vss
558 }%
559 \fi
560 }
561 </plcore|latexrelease>
562 <latexrelease>\plEndIncludeInRelease
563 <latexrelease>\plIncludeInRelease{2016/04/17}{\@outputtombow}
564 <latexrelease> {Safe \boxmaxdepth}%
565 <latexrelease>\def\@outputtombow{%
566 <latexrelease> \iftombow
567 <latexrelease> \vbox to\z@{\kern-13mm\relax
568 <latexrelease> \boxmaxdepth\maxdimen
569 <latexrelease> \moveleft3mm\vbox to\@@paperheight{%
570 <latexrelease> \hbox to\@@paperwidth{\hskip3mm\relax
571 <latexrelease> \copy\@TL\hfill\copy\@TC\hfill\copy\@TR\hskip3mm}%
572 <latexrelease> \kern-10mm
573 <latexrelease> \hbox to\@@paperwidth{\copy\@Tl\hfill\copy\@Tr}%
574 <latexrelease> \vfill
575 <latexrelease> \hbox to\@@paperwidth{\copy\@CL\hfill\copy\@CR}%
576 <latexrelease> \vfill
577 <latexrelease> \hbox to\@@paperwidth{\copy\@Bl\hfill\copy\@Br}%
578 <latexrelease> \kern-10mm
579 <latexrelease> \hbox to\@@paperwidth{\hskip3mm\relax
580 <latexrelease> \copy\@BL\hfill\copy\@BC\hfill\copy\@BR\hskip3mm}%
581 <latexrelease> }\vss
582 <latexrelease> }%
583 <latexrelease> \fi
584 <latexrelease>}
585 <latexrelease>\plEndIncludeInRelease
586 <latexrelease>\plIncludeInRelease{0000/00/00}{\@outputtombow}
587 <latexrelease> {ASCII Corporation original}%
588 <latexrelease>\def\@outputtombow{%
589 <latexrelease> \iftombow
590 <latexrelease> \vbox to\z@{\kern-13mm\relax
591 <latexrelease> \moveleft3mm\vbox to\@@paperheight{%
592 <latexrelease> \hbox to\@@paperwidth{\hskip3mm\relax
593 <latexrelease> \copy\@TL\hfill\copy\@TC\hfill\copy\@TR\hskip3mm}%
594 <latexrelease> \kern-10mm
595 <latexrelease> \hbox to\@@paperwidth{\copy\@Tl\hfill\copy\@Tr}%
596 <latexrelease> \vfill
597 <latexrelease> \hbox to\@@paperwidth{\copy\@CL\hfill\copy\@CR}%
598 <latexrelease> \vfill
599 <latexrelease> \hbox to\@@paperwidth{\copy\@Bl\hfill\copy\@Br}%
600 <latexrelease> \kern-10mm
601 <latexrelease> \hbox to\@@paperwidth{\hskip3mm\relax

```

```

602 <latexrelease>          \copy\@BL\hfill\copy\@BC\hfill\copy\@BR\hskip3mm}%
603 <latexrelease>      }\vss
604 <latexrelease>  }%
605 <latexrelease>  \fi
606 <latexrelease>}
607 <latexrelease>\plEndIncludeInRelease
608 <*plcore>

```

\@@paperheight \@@paperheight は、用紙の縦の長さにトンボの長さを加えた長さになります。

\@@paperwidth \@@paperwidth は、用紙の横の長さにトンボの長さを加えた長さになります。

\@@topmargin \@@topmargin は、現在のトップマージンに 1 インチ加えた長さになります。

```

609 \newdimen\@@paperheight
610 \newdimen\@@paperwidth
611 \newdimen\@@topmargin

```

\@tombowreset@@paper トンボ出力オプションが指定されている場合に用紙サイズを再設定する命令です。

\@outputpage へ加える変更を簡潔にするため、分離した上で \@tombowbleed を使うようにしました。

```

612 </plcore>
613 <latexrelease>\plIncludeInRelease{2018/05/20}{\@tombowreset@@paper}
614 <latexrelease>          {Macro separated}%
615 <*plcore | latexrelease>
616 \def\@tombowreset@@paper{%
617     \@@topmargin\topmargin
618     \iftombow
619         \@@paperwidth\paperwidth
620         \advance\@@paperwidth 2\dimexpr\@tombowbleed\relax
621         \@@paperheight\paperheight \advance\@@paperheight 10mm\relax
622         \advance\@@paperheight 2\dimexpr\@tombowbleed\relax
623         \advance\@@topmargin 1in\relax \advance\@themargin 1in\relax
624     \fi
625 }
626 </plcore | latexrelease>
627 <latexrelease>\plEndIncludeInRelease
628 <latexrelease>\plIncludeInRelease{0000/00/00}{\@tombowreset@@paper}
629 <latexrelease>          {Macro separated}%
630 <latexrelease>\let\@tombowreset@@paper\undefined
631 <latexrelease>\plEndIncludeInRelease
632 <*plcore>

```

11.7 出力ルーチン

ここで実際にトンボを出力します。

\@shipoutsetup \@outputpage 内に挿入したので削除しました。

\@outputpage \textwidth と \textheight の交換は、\@shipoutsetup 内では行ないません。なぜなら、\@shipoutsetup マクロが実行されるときは、\shipout される \vbox の中であり、このときは横組モードですので、つねに \iftdir は偽と判断され、縦と横のサイズを交換できないからです。

なお、この変更をローカルなものにするために、`\begingroup` と `\endgroup` で囲みます。

```
633 \plcore
634 \plincludeinrelease{2018/05/20}{\@outputpage}
635 \plcore \plcore \plcore {Use \@tombowreset@paper}%
636 \def\@outputpage{%
637 \begingroup % the \endgroup is put in by \aftergroup
638 \iftdir
639 \dimen\z@ \textwidth \textwidth \textheight \textheight \dimen\z@
640 \fi
641 \let \protect \noexpand
```

L^AT_EX 2_ε 2017-04-15 では verbatim 環境内でハイフネーションが起きないように修正されましたが、verbatim 環境の途中で改ページが起きた場合にヘッダでハイフネーションが抑制されるのは正しくないので、`\language` を `\begin{document}` で の値にリセットします（参考：latex2e svn r1407）。プリアンプルで特別に設定されればその値、設定されなければ 0 です（万が一 `\document` の定義が古い場合³は -1 になりますが、これは 0 と同じはたらきをするので問題は起きません）。

```
643 \language\document@default@language
644 \@resetactivechars
645 \global\let\@if@newlist@if@newlist
646 \global\@newlistfalse
647 \@parboxrestore
648 \shipout\vbox{\yoko
649 \set@typeset@protect
650 \aftergroup\endgroup
651 \aftergroup\set@typeset@protect
```

ここから `\@shipoutsetup` の内容。

```
652 \if@specialpage
653 \global\@specialpagefalse\@nameuse{ps@\@specialstyle}%
654 \fi

655 \if@twoside
656 \ifodd\count\z@ \let\@thehead\@oddhead \let\@thefoot\@oddfoot
657 \iftdir\let\@themargin\evensidemargin
658 \else\let\@themargin\oddsidemargin\fi
659 \else \let\@thehead\@evenhead
660 \let\@thefoot\@evenfoot
661 \iftdir\let\@themargin\oddsidemargin
662 \else\let\@themargin\evensidemargin\fi
663 \fi\fi
```

トンボ出力オプションが指定されている場合、ここで用紙サイズを再設定します。T_EX の加える左と上部の 1 インチは、トンボの内側に入ります。

```
664 \@tombowreset@paper
665 \reset@font
666 \normalsize
```

³L^AT_EX 2_ε 2017/01/01 以前を使って pL^AT_EX 2_ε のフォーマットを作成した場合や、dinbrief.cls のように独自の再定義を行うクラスやパッケージを使った場合に起こるかもしれません。

```

667 \normalsfcodes
668 \let\label\@gobble
669 \let\index\@gobble
670 \let\glossary\@gobble
671 \baselineskip\z@skip \lineskip\z@skip \lineskiplimit\z@

```

ここまでが \@shipoutsetup の内容。

```

672 \@beginndvi
673 \@outputtombow
674 \vskip \@@topmargin
675 \moveright\@themargin\ vbox{%
676   \setbox\@tempboxa \vbox to\headheight{%
677     \vfil
678     \color@hbox
679     \normalcolor
680     \hb@xt@\textwidth{\@thehead}%
681     \color@endbox
682   }%
683   \dp\@tempboxa \z@
684   \box\@tempboxa
685   \vskip \headsep
686   \box\@outputbox
687   \baselineskip \footskip
688   \color@hbox
689   \normalcolor
690   \hb@xt@\textwidth{\@thefoot}%
691   \color@endbox
692 }%
693 }%
694 % \endgroup now inserted by \aftergroup

\if@newlist を初期化。
695 \global\let\if@newlist\@if@newlist
696 \global \colht \textheight
697 \stepcounter{page}%
698 \let\firstmark\botmark
699 }
700 </plcore | platexrelease>
701 <platexrelease>\plEndIncludeInRelease
702 <platexrelease>\plIncludeInRelease{2017/04/08}{\@outputpage}
703 <platexrelease> {Reset language for hyphenation}%
704 <platexrelease>\def\@outputpage{%
705 <platexrelease>\begingroup % the \endgroup is put in by \aftergroup
706 <platexrelease> \iftdir
707 <platexrelease> \dimen\z@\textwidth \textwidth\textheight \textheight\dimen\z@
708 <platexrelease> \fi
709 <platexrelease> \let \protect \noexpand
710 <platexrelease> \language\document@default@language
711 <platexrelease> \@resetactivechars
712 <platexrelease> \global\let\@if@newlist\if@newlist
713 <platexrelease> \global\@newlistfalse
714 <platexrelease> \@parboxrestore
715 <platexrelease> \shipout\vbox{\yoko
716 <platexrelease> \set@typeset@protect

```

```

717 <platexrelease> \aftergroup\endgroup
718 <platexrelease> \aftergroup\set@typeset@protect
719 <platexrelease> \if@specialpage
720 <platexrelease> \global\@specialpagefalse\@nameuse{ps@\@specialstyle}%
721 <platexrelease> \fi
722 <platexrelease> \if@twoside
723 <platexrelease> \ifodd\count\z@ \let\@thehead\@oddhead \let\@thefoot\@oddfoot
724 <platexrelease> \iftdir\let\@themargin\evensidemargin
725 <platexrelease> \else\let\@themargin\oddsidemargin\fi
726 <platexrelease> \else \let\@thehead\@evenhead
727 <platexrelease> \let\@thefoot\@evenfoot
728 <platexrelease> \iftdir\let\@themargin\oddsidemargin
729 <platexrelease> \else\let\@themargin\evensidemargin\fi
730 <platexrelease> \fi\fi
731 <platexrelease> \@@topmargin\topmargin
732 <platexrelease> \iftombow
733 <platexrelease> \@@paperwidth\paperwidth \advance\@@paperwidth 6mm\relax
734 <platexrelease> \@@paperheight\paperheight \advance\@@paperheight 16mm\relax
735 <platexrelease> \advance\@@topmargin 1in\relax \advance\@themargin 1in\relax
736 <platexrelease> \fi
737 <platexrelease> \reset@font
738 <platexrelease> \normalsize
739 <platexrelease> \normalsfcodes
740 <platexrelease> \let\label\@gobble
741 <platexrelease> \let\index\@gobble
742 <platexrelease> \let\glossary\@gobble
743 <platexrelease> \baselineskip\z@skip \lineskip\z@skip \lineskiplimit\z@
744 <platexrelease> \@begindvi
745 <platexrelease> \@outputtombow
746 <platexrelease> \vskip \@@topmargin
747 <platexrelease> \moveright\@themargin\vbox{%
748 <platexrelease> \setbox\@tempboxa \vbox to\headheight{%
749 <platexrelease> \vfil
750 <platexrelease> \color@hbox
751 <platexrelease> \normalcolor
752 <platexrelease> \hb@xt@\textwidth{\@thehead}%
753 <platexrelease> \color@endbox
754 <platexrelease> }% %% 22 Feb 87
755 <platexrelease> \dp\@tempboxa \z@
756 <platexrelease> \box\@tempboxa
757 <platexrelease> \vskip \headsep
758 <platexrelease> \box\@outputbox
759 <platexrelease> \baselineskip \footskip
760 <platexrelease> \color@hbox
761 <platexrelease> \normalcolor
762 <platexrelease> \hb@xt@\textwidth{\@thefoot}%
763 <platexrelease> \color@endbox
764 <platexrelease> }%
765 <platexrelease> }%
766 <platexrelease> \global\let\if@newlist\@if@newlist
767 <platexrelease> \global \colht \textheight
768 <platexrelease> \stepcounter{page}%
769 <platexrelease> \let\firstmark\botmark
770 <platexrelease> }

```

```

771 <latexrelease>\plEndIncludeInRelease
772 <latexrelease>\plIncludeInRelease{0000/00/00}{\@outputpage}
773 <latexrelease>          {ASCII Corporation original}%
774 <latexrelease>\def\@outputpage{%
775 <latexrelease>\begingroup % the \endgroup is put in by \aftergroup
776 <latexrelease> \iftdir
777 <latexrelease>   \dimen\z@\textwidth \textwidth\textheight \textheight\dimen\z@
778 <latexrelease> \fi
779 <latexrelease> \let \protect \noexpand
780 <latexrelease> \@resetactivechars
781 <latexrelease> \global\let\@@if@newlist@if@newlist
782 <latexrelease> \global\@newlistfalse
783 <latexrelease> \@parboxrestore
784 <latexrelease> \shipout\vbox{\yoko
785 <latexrelease>   \set@typeset@protect
786 <latexrelease>   \aftergroup\endgroup
787 <latexrelease>   \aftergroup\set@typeset@protect
788 <latexrelease>   \if@specialpage
789 <latexrelease>     \global\@specialpagefalse\@nameuse{ps@\@specialstyle}%
790 <latexrelease>   \fi
791 <latexrelease>   \if@twoside
792 <latexrelease>     \ifodd\count\z@ \let\@thehead\@oddhead \let\@thefoot\@oddfoot
793 <latexrelease>     \iftdir\let\@themargin\evensidemargin
794 <latexrelease>     \else\let\@themargin\oddsidemargin\fi
795 <latexrelease>   \else \let\@thehead\@evenhead
796 <latexrelease>     \let\@thefoot\@evenfoot
797 <latexrelease>     \iftdir\let\@themargin\oddsidemargin
798 <latexrelease>     \else\let\@themargin\evensidemargin\fi
799 <latexrelease>   \fi\fi
800 <latexrelease>   \@@topmargin\topmargin
801 <latexrelease>   \iftombow
802 <latexrelease>     \@@paperwidth\paperwidth \advance\@@paperwidth 6mm\relax
803 <latexrelease>     \@@paperheight\paperheight \advance\@@paperheight 16mm\relax
804 <latexrelease>     \advance\@@topmargin 1in\relax \advance\@themargin 1in\relax
805 <latexrelease>   \fi
806 <latexrelease>   \reset@font
807 <latexrelease>   \normalsize
808 <latexrelease>   \normalsfcodes
809 <latexrelease>   \let\label\@gobble
810 <latexrelease>   \let\index\@gobble
811 <latexrelease>   \let\glossary\@gobble
812 <latexrelease>   \baselineskip\z@skip \lineskip\z@skip \lineskiplimit\z@
813 <latexrelease> \@beginvdi
814 <latexrelease> \@outputtombow
815 <latexrelease> \vskip \@@topmargin
816 <latexrelease> \moveright\@themargin\vbox{%
817 <latexrelease>   \setbox\@tempboxa \vbox to\headheight{%
818 <latexrelease>     \vfil
819 <latexrelease>     \color@hbox
820 <latexrelease>       \normalcolor
821 <latexrelease>         \hb@xt@\textwidth{\@thehead}%
822 <latexrelease>       \color@endbox
823 <latexrelease>     }%
824 <latexrelease>   \dp\@tempboxa \z@

```

```

825 <platexrelease> \box\@tempboxa
826 <platexrelease> \vskip \headsep
827 <platexrelease> \box\@outputbox
828 <platexrelease> \baselineskip \footskip
829 <platexrelease> \color@hbox
830 <platexrelease> \normalcolor
831 <platexrelease> \hb@xt@\textwidth{\@thefoot}%
832 <platexrelease> \color@endbox
833 <platexrelease> }%
834 <platexrelease> }%
835 <platexrelease> \global\let\if@newlist\@if@newlist
836 <platexrelease> \global \colht \textheight
837 <platexrelease> \stepcounter{page}%
838 <platexrelease> \let\firstmark\botmark
839 <platexrelease> }
840 <platexrelease> \plEndIncludeInRelease
841 <*plcore>

```

\AtBeginDvi L^AT_EX 2_ε のバージョンが 2020-02-02 までの場合：\AtBeginDvi が「\unvbox してから再び \vbox する」という動作のため、再定義が必要です。

pL^AT_EX の出力ルーチンの \@outputpage では、\shipout する vbox の中に \yoko を指定しています。このため、\AtBeginDocument{\AtBeginDvi{}} というコードを書くと Incompatible direction list can't be unboxed. というエラーが出てしまいます。

そこで、コミュニティ版 pL^AT_EX では「\shipout で \yoko が指定されている」ことを根拠として

\@begindvibox は（空でない限り）常に横組でなければならない

と仮定します。この仮定に従い、\AtBeginDvi を再定義します。

L^AT_EX 2_ε 2020-10-01 以降：\AtBeginDvi はフックにどんどんコードを追加していくだけです。一方、代わりに __shipout_execute_cont: を再定義する必要があります。

```

842 </plcore>
843 <platexrelease> \plIncludeInRelease{2020/10/01}{\AtBeginDvi}
844 <platexrelease> {Adapt to new shipout code}%
845 <*plcore | platexrelease>
846 \chardef\pltx@AtBeginDvi@untouched\z@
847 % for LaTeX2e 2020-10-01 or later
848 \ifdefined\ExplSyntaxOn %--- expl3 available BEGIN
849 \ExplSyntaxOn
850 \if_cs_exist:N \__shipout_add_firstpage_material:Nn
851 \chardef\pltx@AtBeginDvi@untouched\@ne
852 <platexrelease> \DeclareRobustCommand \AtBeginDvi
853 <platexrelease> {\__shipout_add_firstpage_material:Nn \AtBeginDvi}
854 \fi:
855 \ExplSyntaxOff
856 \fi %--- expl3 available END

```

```

857 % for LaTeX2e 2020-02-02 PL5 or older
858 \ifnum\pltx@AtBeginDvi@untouched>z@\else
859 \DeclareRobustCommand \AtBeginDvi [1]{%
860   \global \setbox \@begindvibox
861     \vbox{\yoko \unvbox \@begindvibox #1}}%
862 \fi
863 % done
864 \let\pltx@AtBeginDvi@untouched\@undefined
865 </plcore| platexrelease>
866 <platexrelease>\plEndIncludeInRelease
867 <platexrelease>\plIncludeInRelease{2019/10/01}{\AtBeginDvi}
868 <platexrelease>           {Make robust}%
869 <platexrelease>\DeclareRobustCommand \AtBeginDvi [1]{%
870 <platexrelease>   \global \setbox \@begindvibox
871 <platexrelease>     \vbox{\yoko \unvbox \@begindvibox #1}}
872 <platexrelease>\plEndIncludeInRelease
873 <platexrelease>\plIncludeInRelease{2016/07/01}{\AtBeginDvi}
874 <platexrelease>           {Fix for incompatible direction}%
875 <platexrelease>\def \AtBeginDvi #1{%
876 <platexrelease>   \global \setbox \@begindvibox
877 <platexrelease>     \vbox{\yoko \unvbox \@begindvibox #1}}
878 <platexrelease>\expandafter \let \csname AtBeginDvi \endcsname \@undefined
879 <platexrelease>\plEndIncludeInRelease
880 <platexrelease>\plIncludeInRelease{0000/00/00}{\AtBeginDvi}
881 <platexrelease>           {LaTeX2e original}%
882 <platexrelease>\def \AtBeginDvi #1{%
883 <platexrelease>   \global \setbox \@begindvibox
884 <platexrelease>     \vbox{\unvbox \@begindvibox #1}}
885 <platexrelease>\expandafter \let \csname AtBeginDvi \endcsname \@undefined
886 <platexrelease>\plEndIncludeInRelease
887 <*plcore>

```

_shipout_execute_cont: LaTeX 2_ε 2020-10-01 以降: ltshipout.dtx がベースです。ただし、縦組クラスでも通るようにするため、以下の方法を採用します。

- \shipout 実行時の組方向が横組なら、\yoko を実行せずそのまま。
- 横組でない場合は _shipout_execute_cont: を横組ボックス \l_platex_shipout_dummy_box で括って実行する (※)。
- \l_shipout_box が横組でない場合は事前に横組化する。

注意: 上記※の実装により、縦組クラスでは「\AtBeginShipout の中身が外部垂直モードで実行されること」を想定した使用 (例: platex-tools issue #15) はサポートされません。

```

888 </plcore>
889 <platexrelease>\plIncludeInRelease{2020/10/01}{\_shipout\_execute\_cont:}
890 <platexrelease>           {Adapt to new shipout code}%
891 <*plcore| platexrelease>
892 \ifdefined\ExplSyntaxOn %--- expl3 available BEGIN
893 \ExplSyntaxOn
894 \cs_if_exist:NT \_shipout\_execute\_cont: { %--- IF LEVEL 1 BEGIN

```



```

895 \cs_if_exist:NF \__platex_original_shipout_execute_cont: {
896   \cs_new_eq:NN \__platex_original_shipout_execute_cont:
897       \__shipout_execute_cont:
898 }
899 \cs_if_exist:NF \l__platex_shipout_dummy_box {
900   \box_new:N \l__platex_shipout_dummy_box
901 }
902 \cs_set:Npn \__shipout_execute_cont:
903 {
904   % if \l_shipout_box is not a \yoko-box (= horizontal writing),
905   % then make it a \yoko-box behorehand.
906   \platex_if_box_yoko:NF \l_shipout_box {
907     \vbox_set:Nn \l_shipout_box
908     {
909       \platex_direction_yoko:
910       \box_use:N \l_shipout_box
911     }
912   }
913   % if the current direction is not \yoko,
914   % then enclose \__shipout_execute_cont: with
915   % a dummy \yoko-box named \l__platex_shipout_dummy_box.
916   \platex_if_direction_yoko:TF {
917     \__platex_original_shipout_execute_cont:
918   }{
919     \vbox_set:Nn \l__platex_shipout_dummy_box
920     {
921       \platex_direction_yoko:
922       \__platex_original_shipout_execute_cont:
923     }
924     % [Limitation] the code above may discard some contents,
925     % so we'd like to put it back by \box\l__platex_shipout_dummy_box.
926     % however, an infinite loop occurs if we uncomment the line below
927     % so we can't.
928     %\box_use:N \l__platex_shipout_dummy_box
929   }
930 }
931 }

```

%--- IF LEVEL 1 END

`_shipout_execute_nohooks_cont:` L^AT_EX 2_ε 2021-06-01 では、同様の処理が `__shipout_execute_nohooks_cont:` にも必要なので、それを行います。

```

932 \cs_if_exist:NT \__shipout_execute_nohooks_cont: { %--- IF LEVEL 1 BEGIN
933   \cs_if_exist:NF \__platex_original_shipout_execute_nohooks_cont: {
934     \cs_new_eq:NN \__platex_original_shipout_execute_nohooks_cont:
935         \__shipout_execute_nohooks_cont:
936   }
937   \cs_set:Npn \__shipout_execute_nohooks_cont:
938   {
939     \platex_if_box_yoko:NF \l__shipout_raw_box {
940       \vbox_set:Nn \l__shipout_raw_box
941       {
942         \platex_direction_yoko:
943         \box_use:N \l__shipout_raw_box
944       }
945     }
946   }

```

```

946 \platex_if_direction_yoko:TF {
947   \__platex_original_shipout_execute_nohooks_cont:
948 }{
949   \vbox_set:Nn \l__platex_shipout_dummy_box
950   {
951     \platex_direction_yoko:
952     \__platex_original_shipout_execute_nohooks_cont:
953   }
954 }
955 }
956 } %--- IF LEVEL 1 END
957 \ExplSyntaxOff
958 \fi %--- expl3 available END
959 </plcore| platexrelease>
960 <platexrelease>\plEndIncludeInRelease
961 <platexrelease>\plIncludeInRelease{0000/00/00}{\__shipout_execute_cont:}
962 <platexrelease> {LaTeX2e original}%
963 <platexrelease>% do nothing
964 <platexrelease>\plEndIncludeInRelease

```

11.8 フロート環境

図 (figure) や表 (table) といったフロート環境内では組方向変更を `\tate` や `\yoko` によって直接変更しないでください。代わりに `plext` パッケージが提供する `\layoutcaption` / `\layoutfloat` / `\pcaption` といった命令を使うと、例えば「縦書き中に横書きで図表キャプションを挿入」のような自由な指定ができるようになります。(Issue #95)

`\@floatboxreset` 元は `ltfloat.dtx` で定義されています。

```

965 <platexrelease>\plIncludeInRelease{2023/02/14}{\@floatboxreset}
966 <platexrelease> {Mention \layoutcaption}%
967 <*plcore| platexrelease>
968 \def\@floatboxreset{%
969   \pltx@save@float@dir % pLaTeX
970   \reset@font
971   \normalsize
972   \@setminipage
973 }
974 </plcore| platexrelease>
975 <platexrelease>\plEndIncludeInRelease
976 <platexrelease>\plIncludeInRelease{0000/00/00}{\@floatboxreset}
977 <platexrelease> {Mention \layoutcaption}%
978 <platexrelease>\def\@floatboxreset{%
979 <platexrelease> \reset@font
980 <platexrelease> \normalsize
981 <platexrelease> \@setminipage
982 <platexrelease>}
983 <platexrelease>\plEndIncludeInRelease

```

`\@endfloatbox` 元は `ltfloat.dtx` で定義されています。

```

984 <platexrelease>\plIncludeInRelease{2023/02/14}{\@endfloatbox}

```

```

985 <latexrelease> {Mention \layoutcaption}%
986 <*plcore | latexrelease>
987 \def\@endfloatbox{%
988     \par\vskip\z@skip %% \par\vskip\z@ added 15 Dec 87
989     \minipagefalse
990     \outer@nobreak
991     \pltx@check@float@dir % pLaTeX
992     \egroup %% end of vbox
993     \color@endbox
994 }
995 </plcore | latexrelease>
996 <latexrelease>\plEndIncludeInRelease
997 <latexrelease>\plIncludeInRelease{0000/00/00}{\@endfloatbox}
998 <latexrelease> {Mention \layoutcaption}%
999 <latexrelease>\def\@endfloatbox{%
1000 <latexrelease> \par\vskip\z@skip %% \par\vskip\z@ added 15 Dec 87
1001 <latexrelease> \minipagefalse
1002 <latexrelease> \outer@nobreak
1003 <latexrelease> \egroup %% end of vbox
1004 <latexrelease> \color@endbox
1005 <latexrelease>}
1006 <latexrelease>\plEndIncludeInRelease

```

\pltx@save@float@dir pLaTeX による追加命令です。

```

\pltx@check@float@dir007 <latexrelease>\plIncludeInRelease{2023/02/14}{\pltx@check@float@dir}
1008 <latexrelease> {Mention \layoutcaption}%
1009 <*plcore | latexrelease>
1010 \def\pltx@save@float@dir{%
1011     \edef\pltx@float@dir@first{\iftdir\tate\else\yoko\fi}
1012     \def\pltx@check@float@dir{%
1013         \ifx\pltx@float@dir@first\@undefined\else
1014             \edef\pltx@float@dir@last{\iftdir\tate\else\yoko\fi}%
1015             \ifx\pltx@float@dir@last\pltx@float@dir@first\else
1016                 \pltx@err@float@dir
1017             \fi
1018         \fi}
1019 \def\pltx@err@float@dir{%
1020     \@latex@error{Direction change inside float!?\MessageBreak
1021     Use \noexpand\layoutcaption provided in
1022     \string\usepackage{plext}}\@ehc}
1023 </plcore | latexrelease>
1024 <latexrelease>\plEndIncludeInRelease
1025 <latexrelease>\plIncludeInRelease{0000/00/00}{\pltx@check@float@dir}
1026 <latexrelease> {Mention \layoutcaption}%
1027 <latexrelease>\let\pltx@save@float@dir\@undefined
1028 <latexrelease>\let\pltx@check@float@dir\@undefined
1029 <latexrelease>\let\pltx@err@float@dir\@undefined
1030 <latexrelease>\plEndIncludeInRelease
1031 <*plcore>

```

11.9 脚注マクロ

脚注を組み立てる部分のマクロを再定義します。主な修正点は、縦組モードでの動作の追加です。

これらのマクロは、`ltfloat.dtx` で定義されていたものです。

`\thempfn` 本文で使われる脚注記号です。

`\@footnotemark` で縦横の判断をするようにしたため、削除。

```
1032 %\def\thempfn{%
1033 % \ifdir\thefootnote\else\hbox{\yoko\thefootnote}\fi}
```

`\thempfootnote` minipage 環境で使われる脚注記号です。

```
1034 %\def\thempfootnote{%
1035 % \ifdir\alph{mpfootnote}\else\hbox{\yoko\alph{mpfootnote}}\fi}
```

`\@makefnmark` 脚注記号を作成するマクロです。

```
1036 </plcore>
1037 <latexrelease>\plIncludeInRelease{2016/04/17}{\@makefnmark}
1038 <latexrelease> {Remove extra \xkanjiskip}%
1039 <*plcore | latexrelease>
1040 \renewcommand\@makefnmark{%
1041 \ifdir \hbox{\hbox{\@textsuperscript{\normalfont\@thefnmark}}\hbox{}}%
1042 \else\hbox{\yoko\@textsuperscript{\normalfont\@thefnmark}}\fi}
1043 </plcore | latexrelease>
1044 <latexrelease>\plEndIncludeInRelease
1045 <latexrelease>\plIncludeInRelease{0000/00/00}{\@makefnmark}
1046 <latexrelease> {ASCII Corporation original}%
1047 <latexrelease>\renewcommand\@makefnmark{\hbox{%
1048 <latexrelease> \ifdir \@textsuperscript{\normalfont\@thefnmark}%
1049 <latexrelease> \else\hbox{\yoko\@textsuperscript{\normalfont\@thefnmark}}\fi}}
1050 <latexrelease>\plEndIncludeInRelease
```

`\pltx@foot@penalty` 開き括弧類の直後に `\footnotetext` が続いた場合、`\footnotetext` の前での改行は望ましくありません。このような場合に対処するために、`\pltx@foot@penalty` というカウンタを用意しました。`\footnotetext` の最初で「直前のペナルティ値」としてこのカウンタが初期化されます。`\footnotemark`, `\footnote` では使わないので 0 に設定しています。

```
1051 <latexrelease>\plIncludeInRelease{2016/09/03}{\pltx@foot@penalty}
1052 <latexrelease> {Add new counter \pltx@foot@penalty}%
1053 <*plcore | latexrelease>
1054 \ifx\@undefined\pltx@foot@penalty \newcount\pltx@foot@penalty \fi
1055 \pltx@foot@penalty\z@
1056 </plcore | latexrelease>
1057 <latexrelease>\plEndIncludeInRelease
1058 <latexrelease>\plIncludeInRelease{0000/00/00}{\pltx@foot@penalty}
1059 <latexrelease> {Add new counter \pltx@foot@penalty}%
1060 <latexrelease>\let\pltx@foot@penalty\@undefined
1061 <latexrelease>\plEndIncludeInRelease
```

`\footnotemark` また、合印の前の文字と合印の間は原則ベタ組です (但し、JIS X 4051 には例外有り)。

`\footnote` そのため、合印を出力する `\footnotemark`, `\footnote` の最初で `\inhibitglue` を実行しておくことにします (`\@makefnmark` の中に置いても効力がありません)。

```
1062 <latexrelease>\plIncludeInRelease{2016/09/03}{\footnote}
1063 <latexrelease>                                {Append \inhibitglue in \footnotemark}%
1064 <*plcore | latexrelease>
1065 \def\footnote{\inhibitglue
1066     \@ifnextchar[\@xfootnote{\stepcounter\@mpfn
1067     \protected@xdef\@thefnmark{\thempfn}%
1068     \@footnotemark\@footnotetext}}
1069 \def\footnotemark{\inhibitglue
1070     \@ifnextchar[\@xfootnotemark
1071     {\stepcounter{footnote}%
1072     \protected@xdef\@thefnmark{\thefootnote}%
1073     \@footnotemark}}
1074 </plcore | latexrelease>
1075 <latexrelease>\plEndIncludeInRelease
1076 <latexrelease>\plIncludeInRelease{0000/00/00}{\footnote}
1077 <latexrelease>                                {LaTeX2e original}%
1078 <latexrelease>\def\footnote{\@ifnextchar[\@xfootnote{\stepcounter\@mpfn
1079 <latexrelease>     \protected@xdef\@thefnmark{\thempfn}%
1080 <latexrelease>     \@footnotemark\@footnotetext}}
1081 <latexrelease>\def\footnotemark{%
1082 <latexrelease>     \@ifnextchar[\@xfootnotemark
1083 <latexrelease>     {\stepcounter{footnote}%
1084 <latexrelease>     \protected@xdef\@thefnmark{\thefootnote}%
1085 <latexrelease>     \@footnotemark}}
1086 <latexrelease>\plEndIncludeInRelease
```

`\footnotetext` `\footnotetext` の直前のペナルティ 値を保持します。

```
1087 <latexrelease>\plIncludeInRelease{2016/09/03}{\footnotetext}
1088 <latexrelease>                                {Preserve penalty before \footnotetext}%
1089 <*plcore | latexrelease>
1090 \def\footnotetext{%
1091     \ifhmode\plt@foot@penalty\lastpenalty\unpenalty\fi%
1092     \@ifnextchar [\@xfootnotenext
1093     {\protected@xdef\@thefnmark{\thempfn}%
1094     \@footnotetext}}
1095 </plcore | latexrelease>
1096 <latexrelease>\plEndIncludeInRelease
1097 <latexrelease>\plIncludeInRelease{0000/00/00}{\footnotetext}
1098 <latexrelease>                                {LaTeX2e original}%
1099 <latexrelease>\def\footnotetext{%
1100 <latexrelease>     \@ifnextchar [\@xfootnotenext
1101 <latexrelease>     {\protected@xdef\@thefnmark{\thempfn}%
1102 <latexrelease>     \@footnotetext}}
1103 <latexrelease>\plEndIncludeInRelease
```

`\@footnotetext` インサートボックス `\footins` に脚注のテキストを入れます。コミュニティ版 p_{La}T_EX では `\footnotetext`, `\footnote` の直後で改行を可能にします。jsclasses ではこの変更に加え、脚注で `\verb` が使えるように再定義されます。

```

1104 <platexrelease>\plIncludeInRelease{2021/11/15}{\@footnotetext}
1105 <platexrelease> {Adapt to ltfloat.dtx (2021-10-14 v1.2g)}%
1106 <*plcore | platexrelease>
1107 \long\def\@footnotetext#1{%
1108   \ifdir\def\@tempa{\yoko}\else\def\@tempa{\tate}\fi
1109   \insert\footins{\@tempa%
1110     \reset@font\footnotesize
1111     \interlinepenalty\interfootnotelinepenalty
1112     \splittopskip\footnotesep
1113     \splitmaxdepth \dp\strutbox \floatingpenalty \@MM
1114     \hsize\columnwidth \@parboxrestore
1115     \def\@currentcounter{footnote}%
1116     \protected@edef\@currentlabel{%
1117       \csname p@footnote\endcsname\@thefnmark
1118     }%
1119     \color@begingroup
1120     \@makefntext{%
1121       \rule{z@\footnotesep\ignorespaces#1\@finalstrut\strutbox}%
1122     \par

```

pTeX では \insert の直後に和文文字が来た場合、そこでの改行は許されないという挙動になっています。このため、従来は脚注番号（合印）の直後の改行が抑制されていました。しかし、\hbox の直後に和文文字が来た場合は、そこでの改行は許されますから、最後に \null を追加します。また、\pltx@foot@penalty の値が 0 でなかった場合、脚注の前にペナルティがあったということですから、復活させておきます。

```

1123   \color@endgroup}\ifhmode\null\fi
1124   \ifnum\pltx@foot@penalty=z@\else
1125     \penalty\pltx@foot@penalty
1126     \pltx@foot@penalty z@
1127   \fi}
1128 </plcore | platexrelease>
1129 <platexrelease>\plEndIncludeInRelease
1130 <platexrelease>\plIncludeInRelease{2021/06/01}{\@footnotetext}
1131 <platexrelease> {Adapt to ltfloat.dtx (2021-03-03 v1.2f)}%
1132 <platexrelease>\long\def\@footnotetext#1{%
1133 <platexrelease> \ifdir\def\@tempa{\yoko}\else\def\@tempa{\tate}\fi
1134 <platexrelease> \insert\footins{\@tempa%
1135 <platexrelease>   \reset@font\footnotesize
1136 <platexrelease>   \interlinepenalty\interfootnotelinepenalty
1137 <platexrelease>   \splittopskip\footnotesep
1138 <platexrelease>   \splitmaxdepth \dp\strutbox \floatingpenalty \@MM
1139 <platexrelease>   \hsize\columnwidth \@parboxrestore
1140 <platexrelease>   \protected@edef\@currentlabel{%
1141 <platexrelease>     \csname p@footnote\endcsname\@thefnmark
1142 <platexrelease>   }%
1143 <platexrelease>   \color@begingroup
1144 <platexrelease>   \@makefntext{%
1145 <platexrelease>     \rule{z@\footnotesep\ignorespaces#1\@finalstrut\strutbox}%
1146 <platexrelease>   \par
1147 <platexrelease>   \color@endgroup}\ifhmode\null\fi
1148 <platexrelease>   \ifnum\pltx@foot@penalty=z@\else

```

```

1149 <platexrelease> \penalty\pltx@foot@penalty
1150 <platexrelease> \pltx@foot@penalty\z@
1151 <platexrelease> \fi}
1152 <platexrelease> \plEndIncludeInRelease
1153 <platexrelease> \plIncludeInRelease{2016/09/08}{\@footnotetext}
1154 <platexrelease> {Allow break after \footnote (more fix)}%
1155 <platexrelease> \long\def\@footnotetext#1{%
1156 <platexrelease> \ifdir\def\@tempa{\yoko}\else\def\@tempa{\tate}\fi
1157 <platexrelease> \insert\footins{\@tempa%
1158 <platexrelease> \reset@font\footnotesize
1159 <platexrelease> \interlinepenalty\interfootnotelinepenalty
1160 <platexrelease> \splittopskip\footnotesep
1161 <platexrelease> \splitmaxdepth \dp\strutbox \floatingpenalty \@MM
1162 <platexrelease> \hsize\columnwidth \@parboxrestore
1163 <platexrelease> \protected@edef\@currentlabel{%
1164 <platexrelease> \csname p@footnote\endcsname\@thefnmark
1165 <platexrelease> }%
1166 <platexrelease> \color@begingroup
1167 <platexrelease> \@makefnintext{%
1168 <platexrelease> \rule\z@\footnotesep\ignorespaces#1\@finalstrut\strutbox}%
1169 <platexrelease> \color@endgroup}\ifhmode\null\fi
1170 <platexrelease> \ifnum\pltx@foot@penalty=\z@\else
1171 <platexrelease> \penalty\pltx@foot@penalty
1172 <platexrelease> \pltx@foot@penalty\z@
1173 <platexrelease> \fi}
1174 <platexrelease> \plEndIncludeInRelease
1175 <platexrelease> \plIncludeInRelease{2016/09/03}{\@footnotetext}
1176 <platexrelease> {Allow break after \footnote}%
1177 <platexrelease> \long\def\@footnotetext#1{%
1178 <platexrelease> \ifdir\def\@tempa{\yoko}\else\def\@tempa{\tate}\fi
1179 <platexrelease> \insert\footins{\@tempa%
1180 <platexrelease> \reset@font\footnotesize
1181 <platexrelease> \interlinepenalty\interfootnotelinepenalty
1182 <platexrelease> \splittopskip\footnotesep
1183 <platexrelease> \splitmaxdepth \dp\strutbox \floatingpenalty \@MM
1184 <platexrelease> \hsize\columnwidth \@parboxrestore
1185 <platexrelease> \protected@edef\@currentlabel{%
1186 <platexrelease> \csname p@footnote\endcsname\@thefnmark
1187 <platexrelease> }%
1188 <platexrelease> \color@begingroup
1189 <platexrelease> \@makefnintext{%
1190 <platexrelease> \rule\z@\footnotesep\ignorespaces#1\@finalstrut\strutbox}%
1191 <platexrelease> \color@endgroup}\null
1192 <platexrelease> \ifnum\pltx@foot@penalty=\z@\else
1193 <platexrelease> \penalty\pltx@foot@penalty
1194 <platexrelease> \pltx@foot@penalty\z@
1195 <platexrelease> \fi}
1196 <platexrelease> \plEndIncludeInRelease
1197 <platexrelease> \plIncludeInRelease{0000/00/00}{\@footnotetext}
1198 <platexrelease> {ASCII Corporation original}%
1199 <platexrelease> \long\def\@footnotetext#1{%
1200 <platexrelease> \ifdir\def\@tempa{\yoko}\else\def\@tempa{\tate}\fi
1201 <platexrelease> \insert\footins{\@tempa%
1202 <platexrelease> \reset@font\footnotesize

```

```

1203 <platexrelease> \interlinepenalty\interfootnotelinepenalty
1204 <platexrelease> \splittopskip\footnotesep
1205 <platexrelease> \splitmaxdepth \dp\strutbox \floatingpenalty \@MM
1206 <platexrelease> \hsize\columnwidth \@parboxrestore
1207 <platexrelease> \protected@edef\@currentlabel{%
1208 <platexrelease> \csname p@footnote\endcsname\@thefnmark
1209 <platexrelease> }%
1210 <platexrelease> \color@begingroup
1211 <platexrelease> \makefnmark{%
1212 <platexrelease> \rule{z@{\footnotesep}\ignorespaces#1\@finalstrut\strutbox}%
1213 <platexrelease> \color@endgroup}}
1214 <platexrelease> \plEndIncludeInRelease
1215 <*plcore>

```

`\@footnotemark` 脚注記号を出力します。

```

1216 \def\@footnotemark{\leavevmode
1217 \ifhmode\edef\@xsf{\the\spacefactor}\nobreak\fi
1218 \ifydir\@makefnmark
1219 \else\hbox to\z@{\hskip-.25zw\raise.9zh\@makefnmark\hss}\fi
1220 \ifhmode\spacefactor\@xsf\fi\relax}

```

11.10 相互参照

`\@setref` `\ref` コマンドや `\pageref` コマンドで参照したとき、これらのコマンドによって出力された番号と続く 2 バイト文字との間に `\xkanjiskip` が入りません。これは、`\null` が `\hbox{}` と定義されているためです。そこで `\null` を取り除きます。このコマンドは、`ltxref.dtx` で定義されているものです。

しかし、単に `\null` を `\relax` に置き換えるだけでは、`\section` のような「動く引数」で `\ref` などを使った場合に、目次で後ろの空白が消えてしまいます。そこで、`\relax` のあとに `\hbox{}` を追加しました。従来も `\protect\ref` のように使えば問題ありませんでしたが、 $\mathrm{\LaTeX}$ では展開されても問題が起きない robust な実装になっていますので、これに従います。

さらに、例えば “see Appendix A.” のような記述が文末にあり、かつ “A” を相互参照で取得した場合のスペースファクターを補正するため、`\spacefactor\@m{}` に修正しました。これで、“A.” の後のスペースが文末として扱われます。「 $\mathrm{\LaTeX}$ 2_ε マクロ&クラス プログラミング実践解説」のコードを参考にしましたが、数式モード内でもエラーにならないように改良しています。

```

1221 </plcore>
1222 <platexrelease> \plIncludeInRelease{2017/10/28}{\@setref}
1223 <platexrelease> {Space factor after \ref}%
1224 <*plcore| platexrelease>
1225 \def\@setref#1#2#3{%
1226 \ifx#1\relax
1227 \protect\G@refundefinedtrue
1228 \nfss@text{\reset@font\bfseries ??}%
1229 \@latex@warning{Reference ‘#3’ on page \thepage \space
1230 undefined}%
1231 \else

```



```

1232 \expandafter#2#1\protect\@setref@{}% change \null to \protect\@setref@{}
1233 \fi}
1234 \def\@setref@{\ifhmode\spacefactor\@m\fi}
1235 \plcore| platexrelease)
1236 \platexrelease\plEndIncludeInRelease
1237 \platexrelease\plIncludeInRelease{2017/04/08}{\@setref}
1238 \platexrelease {Spacing after \ref in moving arguments}%
1239 \platexrelease\def\@setref#1#2#3{%
1240 \platexrelease \ifx#1\relax
1241 \platexrelease \protect\G@refundefinedtrue
1242 \platexrelease \nfss@text{\reset@font\bfseries ??}%
1243 \platexrelease \latex@warning{Reference ‘#3’ on page \thepage \space
1244 \platexrelease undefined}%
1245 \platexrelease \else
1246 \platexrelease \expandafter#2#1\relax}% change \null to \relax{}
1247 \platexrelease \fi}
1248 \platexrelease\let\@setref@\@undefined
1249 \platexrelease\plEndIncludeInRelease
1250 \platexrelease\plIncludeInRelease{0000/00/00}{\@setref}
1251 \platexrelease {ASCII Corporation original}%
1252 \platexrelease\def\@setref#1#2#3{%
1253 \platexrelease \ifx#1\relax
1254 \platexrelease \protect\G@refundefinedtrue
1255 \platexrelease \nfss@text{\reset@font\bfseries ??}%
1256 \platexrelease \latex@warning{Reference ‘#3’ on page \thepage \space
1257 \platexrelease undefined}%
1258 \platexrelease \else
1259 \platexrelease \expandafter#2#1\relax% change \null to \relax
1260 \platexrelease \fi}
1261 \platexrelease\let\@setref@\@undefined
1262 \platexrelease\plEndIncludeInRelease
1263 (*plcore)

```

11.11 疑似タイプ入力

`\verb` L^AT_EX の `\verb` コマンドでは、数式モードでないときは、`\leavevmode` で水平モードに入ったあと、`\null` を出力しています。マクロ `\null` は `\hbox{}` として定義されていますので、ここには和欧文間スペース (`\xkanjiskip`) が入りません。

しかし、単に `\null` を除いてしまうと、今度は `\verb+ abc+` のように `\verb` の冒頭に半角空白がある場合にこれが消えてしまいます (TeX.SX 170245)。そこで、pL^AT_EX では `\null` の代わりに

1. 和欧文間スペースの挿入処理は透過する
2. 行分割時に消える (discardable) ノードではない

の両条件を満たすノードを挿入します。ここでは `\vadjust{}` としました。

このマクロは、`ltmiscen.dtx` で定義されています。

```

1264 \plcore)
1265 \platexrelease\plIncludeInRelease{2017/10/28}{\verb}
1266 \platexrelease {Preserve beginning space characters}%

```

```

1267 <*plcore | latexrelease>
1268 \if@compatibility\else
1269 \def\verb{\relax\ifmmode\hbox\else\leavevmode\vadjust{}\fi
1270 \bgroup
1271 \verb@eol@error \let\do\@makeother \dospecials
1272 \verbatim@font\@noligs

LaTeX 2ε 2017-04-15 に追従して、\verb の途中でハイフネーションが起きないように
に \language を設定します（参考：latex2e svn r1405）。

1273 \language\l@nohyphenation
1274 \@ifstar\@sverb\@verb}
1275 \fi
1276 </plcore | latexrelease>
1277 <latexrelease>\plEndIncludeInRelease
1278 <latexrelease>\plIncludeInRelease{2017/04/08}{\verb}
1279 <latexrelease> {Disable hyphenation in verb}%
1280 <latexrelease>\if@compatibility\else
1281 <latexrelease>\def\verb{\relax\ifmmode\hbox\else\leavevmode\fi
1282 <latexrelease> \bgroup
1283 <latexrelease> \verb@eol@error \let\do\@makeother \dospecials
1284 <latexrelease> \verbatim@font\@noligs
1285 <latexrelease> \language\l@nohyphenation
1286 <latexrelease> \@ifstar\@sverb\@verb}
1287 <latexrelease>\fi
1288 <latexrelease>\plEndIncludeInRelease
1289 <latexrelease>\plIncludeInRelease{0000/00/00}{\verb}
1290 <latexrelease> {ASCII Corporation original}%
1291 <latexrelease>\if@compatibility\else
1292 <latexrelease>\def\verb{\relax\ifmmode\hbox\else\leavevmode\fi
1293 <latexrelease> \bgroup
1294 <latexrelease> \verb@eol@error \let\do\@makeother \dospecials
1295 <latexrelease> \verbatim@font\@noligs
1296 <latexrelease> \@ifstar\@sverb\@verb}
1297 <latexrelease>\fi
1298 <latexrelease>\plEndIncludeInRelease
1299 <*plcore>

```

\do@noligs >などの一部の文字について、\xspacecode‘\>=3 としたときに \texttt{>}では前後に \xkanjiskip 由来のアキが入るのに、\verb+>+では後ろにしかアキが入らないという現象に対処します。

元の定義は ltmiscen.dtx を参照してください。pLaTeX では、\kern\z@を \vadjust{} に置き換えることで「合字処理を抑止」かつ「和欧文間スペースの挿入処理は透過」を実現します。(Issue #87)

```

1300 </plcore>
1301 <latexrelease>\plIncludeInRelease{2020/04/12}{\do@noligs}
1302 <latexrelease> {Allow \xkanjiskip while avoiding ligature}%
1303 <*plcore | latexrelease>
1304 \def\do@noligs#1{%
1305 \catcode'#1\active
1306 \begingroup
1307 \lccode'\~'#1\relax
1308 \lowercase{\endgroup\def~{\leavevmode\vadjust{}\char'#1}}

```

```

1309 </plcore | latexrelease>
1310 <latexrelease>\plEndIncludeInRelease
1311 <latexrelease>\plIncludeInRelease{0000/00/00}{\do@noligs}
1312 <latexrelease>{LaTeX2e original}%
1313 <latexrelease>\def\do@noligs#1{%
1314 <latexrelease> \catcode'#1\active
1315 <latexrelease> \begingroup
1316 <latexrelease> \lccode'\~'#1\relax
1317 <latexrelease> \lowercase{\endgroup\def~{\leavevmode\kern\z@\char'#1}}
1318 <latexrelease>\plEndIncludeInRelease
1319 <*plcore>

```

11.12 tabbing 環境

`\@startline` tabbing 環境の行で、中身が始め括弧類などで始まる場合、最初の項目だけ JFM グルーが消えない現象に対処します。

```

1320 </plcore>
1321 <latexrelease>\plIncludeInRelease{2017/10/28}{\@startline}
1322 <latexrelease>{Inhibit JFM glue at the beginning}%
1323 <*plcore | latexrelease>
1324 \gdef\@startline{%
1325 \ifnum \@nxttabmar >\@hightab
1326 \badtab
1327 \global\@nxttabmar \@hightab
1328 \fi
1329 \global\@curtabmar \@nxttabmar
1330 \global\@curtab \@curtabmar
1331 \global\setbox\@curline \hbox {}%
1332 \@startfield
1333 \strut\inhibitglue}
1334 </plcore | latexrelease>
1335 <latexrelease>\plEndIncludeInRelease
1336 <latexrelease>\plIncludeInRelease{0000/00/00}{\@startline}
1337 <latexrelease>{LaTeX2e original}%
1338 <latexrelease>\gdef\@startline{%
1339 <latexrelease> \ifnum \@nxttabmar >\@hightab
1340 <latexrelease> \badtab
1341 <latexrelease> \global\@nxttabmar \@hightab
1342 <latexrelease> \fi
1343 <latexrelease> \global\@curtabmar \@nxttabmar
1344 <latexrelease> \global\@curtab \@curtabmar
1345 <latexrelease> \global\setbox\@curline \hbox {}%
1346 <latexrelease> \@startfield
1347 <latexrelease> \strut}
1348 <latexrelease>\plEndIncludeInRelease
1349 <*plcore>

```

`\@stopfield` 相互参照や疑似タイプ入力では、和欧文間スペースが入らないので、`\null`を取り除きましたが、tabbing 環境では、逆に `\null` がないため、和欧文間スペースが入ってしまうので、それを追加します。l1ttab.dtx で定義されているものです。

```

1350 \gdef\@stopfield{\null\color@endgroup\egroup}

```

11.13 用語集の出力

L^AT_EX には、なぜか用語集を出力するためのコマンドがありませんので、追加をします。

`\printglossary` コマンドは、単に拡張子が `gls` のファイルを読み込むだけです。このファイルの生成には、`mendex` などを用います。

```
1351 \newcommand\printglossary{\input@{\jobname.gls}}
```

11.14 時分を示すカウンタ

T_EX には、年月日を示す数値を保持しているカウンタとして、それぞれ `\year`, `\month`, `\day` がプリミティブとして存在します。しかし、時分については、深夜の零時からの経過時間を示す `\time` カウンタしか存在していません。そこで、pL^AT_EX 2_ε では、時分を示すためのカウンタ `\hour` と `\minute` を作成しています。

`\hour` 何時か (`\hour`) を得るには、`\time` を 60 で割った商をそのまま用います。何分か `\minute` (`\minute`) は、`\hour` に 60 を掛けた値を `\time` から引いて算出します。ここではカウンタを宣言するだけです。実際の計算は、クラスやパッケージの中で行なっています。

```
1352 \newcount\hour
1353 \newcount\minute
```

11.15 tabular 環境

L^AT_EX カーネル (`lttab.dtx`) の命令群を修正します。

`\@tabclassz` L^AT_EX カーネルは、アラインメント文字 `&` の周囲に半角空白を書いたかどうかにかかわらず余分なスペースを出力しないように、`\ignorespaces` と `\unskip` を発行しています (`lttab.dtx`)。しかし、これだけでは JFM グルーが消えずに残ってしまうので、pL^AT_EX では追加の対処を入れます。

まず、`l`, `c`, `r` の場合です。2017/09/26 の修正では「セルの要素を `\mbox` に入れ、その最初で `\inhibitglue` を発行する」という方針でしたが、2018/03/09 の修正では「`\removejfmglue` マクロが定義されている場合は最初に `\inhibitglue` を発行し、最後に `\removejfmglue` を発行する」という方針にします。こうすれば少々 L^AT_EX との互換性が向上します。

```
1354 </plcore>
1355 <latexrelease>\plIncludeInRelease{2018/03/09}{\@tabclassz}
1356 <latexrelease>                                {Inhibit JFM glue in tabular cells (better)}%
1357 <*plcore| latexrelease>
1358 \ifx\removejfmglue\@undefined
1359 \def\@tabclassz{%
1360   \ifcase\@lastchclass
1361     \@acolampacol
1362   \or
1363     \@ampacol
```

```

1364 \or
1365 \or
1366 \or
1367 \@addamp
1368 \or
1369 \@acolampacol
1370 \or
1371 \@firstampfalse\@acol
1372 \fi
1373 \edef\@preamble{%
1374 \@preamble{%
1375 \ifcase\@chnum
1376 \hfil\mbox{\inhibitglue\ignorespaces\@sharp\unskip}\hfil % c
1377 \or
1378 \hskip1sp\mbox{\inhibitglue\ignorespaces\@sharp\unskip}\hfil % l
1379 \or
1380 \hfil\hskip1sp\mbox{\inhibitglue\ignorespaces\@sharp\unskip}% % r
1381 \fi}}}
1382 \else
1383 \def\@tabclassz{%
1384 \ifcase\@lastchclass
1385 \@acolampacol
1386 \or
1387 \@ampacol
1388 \or
1389 \or
1390 \or
1391 \@addamp
1392 \or
1393 \@acolampacol
1394 \or
1395 \@firstampfalse\@acol
1396 \fi
1397 \edef\@preamble{%
1398 \@preamble{%
1399 \ifcase\@chnum
1400 \hfil\hskip1sp\inhibitglue
1401 \ignorespaces\@sharp\unskip\removejfmglue\hfil % c
1402 \or
1403 \hskip1sp\inhibitglue
1404 \ignorespaces\@sharp\unskip\removejfmglue\hfil % l
1405 \or
1406 \hfil\hskip1sp\inhibitglue
1407 \ignorespaces\@sharp\unskip\removejfmglue % r
1408 \fi}}}
1409 \fi
1410 </plcore| latexrelease>
1411 <latexrelease>\plEndIncludeInRelease
1412 <latexrelease>\plIncludeInRelease{2017/09/26}{\@tabclassz}
1413 <latexrelease> {Inhibit JFM glue in tabular cells}%
1414 <latexrelease>\def\@tabclassz{%
1415 <latexrelease> \ifcase\@lastchclass
1416 <latexrelease> \@acolampacol
1417 <latexrelease> \or

```

```

1418 <platexrelease> \ampacol
1419 <platexrelease> \or
1420 <platexrelease> \or
1421 <platexrelease> \or
1422 <platexrelease> \addamp
1423 <platexrelease> \or
1424 <platexrelease> \acolampacol
1425 <platexrelease> \or
1426 <platexrelease> \@firstampfalse\acol
1427 <platexrelease> \fi
1428 <platexrelease> \edef\@preamble{%
1429 <platexrelease> \@preamble{%
1430 <platexrelease> \ifcase\@chnum
1431 <platexrelease> \hfil\mbox{\inhibitglue
1432 <platexrelease> \ignorespaces\@sharp\unskip}\hfil % c
1433 <platexrelease> \or
1434 <platexrelease> \hskip1sp\mbox{\inhibitglue
1435 <platexrelease> \ignorespaces\@sharp\unskip}\hfil % l
1436 <platexrelease> \or
1437 <platexrelease> \hfil\hskip1sp\mbox{\inhibitglue
1438 <platexrelease> \ignorespaces\@sharp\unskip}% % r
1439 <platexrelease> \fi}}
1440 <platexrelease> \plEndIncludeInRelease
1441 <platexrelease> \plIncludeInRelease{2017/07/29}{\@tabclassz}
1442 <platexrelease> {Inhibit JFM glue in tabular cells (wrong)}%
1443 <platexrelease> \def\@tabclassz{%
1444 <platexrelease> \ifcase\@lastchclass
1445 <platexrelease> \acolampacol
1446 <platexrelease> \or
1447 <platexrelease> \ampacol
1448 <platexrelease> \or
1449 <platexrelease> \or
1450 <platexrelease> \or
1451 <platexrelease> \addamp
1452 <platexrelease> \or
1453 <platexrelease> \acolampacol
1454 <platexrelease> \or
1455 <platexrelease> \@firstampfalse\acol
1456 <platexrelease> \fi
1457 <platexrelease> \edef\@preamble{%
1458 <platexrelease> \@preamble{%
1459 <platexrelease> \ifcase\@chnum
1460 <platexrelease> \hfil\inhibitglue
1461 <platexrelease> \ignorespaces\@sharp\unskip\unskip\hfil % c
1462 <platexrelease> \or
1463 <platexrelease> \hskip1sp\inhibitglue
1464 <platexrelease> \ignorespaces\@sharp\unskip\unskip\hfil % l
1465 <platexrelease> \or
1466 <platexrelease> \hfil\hskip1sp\inhibitglue
1467 <platexrelease> \ignorespaces\@sharp\unskip\unskip % r
1468 <platexrelease> \fi}}
1469 <platexrelease> \plEndIncludeInRelease
1470 <platexrelease> \plIncludeInRelease{0000/00/00}{\@tabclassz}
1471 <platexrelease> {LaTeX2e original}%

```

```

1472 <platexrelease>\def\@tabclassz{%
1473 <platexrelease> \ifcase\@lastchclass
1474 <platexrelease> \@acolampacol
1475 <platexrelease> \or
1476 <platexrelease> \@ampacol
1477 <platexrelease> \or
1478 <platexrelease> \or
1479 <platexrelease> \or
1480 <platexrelease> \@addamp
1481 <platexrelease> \or
1482 <platexrelease> \@acolampacol
1483 <platexrelease> \or
1484 <platexrelease> \@firstampfalse\@acol
1485 <platexrelease> \fi
1486 <platexrelease> \edef\@preamble{%
1487 <platexrelease> \@preamble{%
1488 <platexrelease> \ifcase\@chnum
1489 <platexrelease> \hfil\ignorespaces\@sharp\unskip\hfil
1490 <platexrelease> \or
1491 <platexrelease> \hskip1sp\ignorespaces\@sharp\unskip\hfil
1492 <platexrelease> \or
1493 <platexrelease> \hfil\hskip1sp\ignorespaces\@sharp\unskip
1494 <platexrelease> \fi}}
1495 <platexrelease>\plEndIncludeInRelease

```

\@classv 次に、p の場合です。2017/07/29 の修正では \mbox{}\inhibitglue と \unskip を追加していましたが、以下のように p 指定のセルの最初で \par として改段落を発行すると、一行空いてしまうという症状が起きてしまいます (latex/#63)。

```

\begin{tabular}{p{5cm}}
A\\
\relax\par
A
\end{tabular}

```

ここでは、2017/07/29 の修正から方針を改め、\everypar 内に \inhibitglue を仕込むという方針で対応します。

```

1496 <platexrelease>\plIncludeInRelease{2018/03/09}{\@classv}
1497 <platexrelease> {Inhibit JFM glue in tabular cells (better)}%
1498 <plcore | platexrelease>
1499 \def\@classv{\@addtopreamble{\@startpbox{\@nextchar}\pltx@next@inhibitglue\ignorespaces
1500 \@sharp\unskip\@endpbox}}
1501 </plcore | platexrelease>
1502 <platexrelease>\plEndIncludeInRelease
1503 <platexrelease>\plIncludeInRelease{2017/07/29}{\@classv}
1504 <platexrelease> {Inhibit JFM glue in tabular cells}%
1505 <platexrelease>\def\@classv{\@addtopreamble{\@startpbox{\@nextchar}\mbox{}\inhibitglue\ignorespaces
1506 <platexrelease>\@sharp\unskip\@endpbox}}
1507 <platexrelease>\plEndIncludeInRelease
1508 <platexrelease>\plIncludeInRelease{0000/00/00}{\@classv}
1509 <platexrelease> {LaTeX2e original}%
1510 <platexrelease>\def\@classv{\@addtopreamble{\@startpbox{\@nextchar}\ignorespaces
1511 <platexrelease>\@sharp\@endpbox}}

```

```
1512 <latexrelease>\plEndIncludeInRelease
```

`\pltx@next@inhibitglue` 水平モードであればそのまま `\inhibitglue` を発行し、それ以外であれば `\everypar` 内に `\inhibitglue` を仕込みます。

```
1513 <latexrelease>\plIncludeInRelease{2018/03/09}{\pltx@next@inhibitglue}
1514 <latexrelease>          {Add \pltx@next@inhibitglue}%
1515 <*plcore | latexrelease>
1516 \protected\def\pltx@next@inhibitglue{%
1517   \ifhmode\inhibitglue\else
1518     \edef\@tempa{\everypar{%
1519       \everypar{\unexpanded\expandafter{\the\everypar}}%
1520       \unexpanded\expandafter{\the\everypar}\inhibitglue}}%
1521   \@tempa\fi}
1522 </plcore | latexrelease>
1523 <latexrelease>\plEndIncludeInRelease
1524 <latexrelease>\plIncludeInRelease{0000/00/00}{\pltx@next@inhibitglue}
1525 <latexrelease>          {Add \pltx@next@inhibitglue}%
1526 <latexrelease>\let\pltx@next@inhibitglue\undefined
1527 <latexrelease>\plEndIncludeInRelease
```

12 2013 年以降の新しい pTeX 対応

LaTeX 2_ε のカーネルのコードをそのまま使うと、2013 年以降の pTeX では `\xkanjiskip` 由来のアキが前後に入ってしまうことがありました。そうした命令にパッチをあてます。なお、既に出てきた `\footnote` の内部命令 (`\@makefnmark`) には同様のパッチがもうあててあります。

`\@tabular` `tabular` 環境の内部命令です。もとは `lttab.dtx` で定義されています。

```
1528 <latexrelease>\plIncludeInRelease{2016/04/17}{\@tabular}
1529 <latexrelease>          {Remove extra \xkanjiskip}%
1530 <*plcore | latexrelease>
1531 \def\@tabular{\leavevmode \null\hbox \bgroup $\let\@acol\@tabacol
1532   \let\@classz\@tabclassz
1533   \let\@classiv\@tabclassiv \let\\\@tabularcr\@tabarray}
1534 </plcore | latexrelease>
1535 <latexrelease>\plEndIncludeInRelease
1536 <latexrelease>\plIncludeInRelease{0000/00/00}{\@tabular}
1537 <latexrelease>          {LaTeX2e original}%
1538 <latexrelease>\def\@tabular{\leavevmode \hbox \bgroup $\let\@acol\@tabacol
1539 <latexrelease>   \let\@classz\@tabclassz
1540 <latexrelease>   \let\@classiv\@tabclassiv \let\\\@tabularcr\@tabarray}
1541 <latexrelease>\plEndIncludeInRelease
```

`\endtabular`

```
\endtabular*1542 <latexrelease>\plIncludeInRelease{2016/04/17}{\endtabular}
1543 <latexrelease>          {Remove extra \xkanjiskip}%
1544 <*plcore | latexrelease>
1545 \def\endtabular{\crrc\egroup\egroup $\egroup\null}
1546 \expandafter \let \csname endtabular*\endcsname = \endtabular
1547 </plcore | latexrelease>
```



```

1548 <latexrelease>\plEndIncludeInRelease
1549 <latexrelease>\plIncludeInRelease{0000/00/00}{\endtabular}
1550 <latexrelease>          {LaTeX2e original}%
1551 <latexrelease>\def\endtabular{\crr\egroup\egroup $\egroup}
1552 <latexrelease>\expandafter \let \csname endtabular*\endcsname = \endtabular
1553 <latexrelease>\plEndIncludeInRelease

```

\@iiiparbox \parbox の内部命令です。もとは ltboxes.dtx で定義されています。

```

1554 <latexrelease>\plIncludeInRelease{2016/04/17}{\@iiiparbox}
1555 <latexrelease>          {Remove extra \xkanjiskip}%
1556 <plcore | latexrelease>
1557 \let \@parboxto \@empty
1558 \long\def \@iiiparbox#1#2[#3]#4#5{%
1559   \leavevmode
1560   \@pboxswfalse
1561   \setlength \@tempdima{#4}%
1562   \@begin@tempboxa\vbox{\hsize \@tempdima \@parboxrestore#5 \@par}%
1563   \ifx \relax#2\else
1564     \setlength \@tempdimb{#2}%
1565     \edef \@parboxto{to\the \@tempdimb}%
1566   \fi
1567   \if#1b\vbox
1568   \else\if #1t\top
1569   \else\ifmmode\center
1570   \else\@pboxswtrue\relax\center% !!!
1571   \fi\fi\fi
1572   \@parboxto{\let \hss \vss \let \unhbox \unvbox
1573     \csname bm@#3\endcsname}%
1574   \if@pboxsw \m@th $\null \fi% !!!
1575   \@end@tempboxa}
1576 </plcore | latexrelease>
1577 <latexrelease>\plEndIncludeInRelease
1578 <latexrelease>\plIncludeInRelease{0000/00/00}{\@iiiparbox}
1579 <latexrelease>          {LaTeX2e original}%
1580 <latexrelease>\let \@parboxto \@empty
1581 <latexrelease>\long\def \@iiiparbox#1#2[#3]#4#5{%
1582 <latexrelease>   \leavevmode
1583 <latexrelease>   \@pboxswfalse
1584 <latexrelease>   \setlength \@tempdima{#4}%
1585 <latexrelease>   \@begin@tempboxa\vbox{\hsize \@tempdima \@parboxrestore#5 \@par}%
1586 <latexrelease>   \ifx \relax#2\else
1587 <latexrelease>     \setlength \@tempdimb{#2}%
1588 <latexrelease>     \edef \@parboxto{to\the \@tempdimb}%
1589 <latexrelease>   \fi
1590 <latexrelease>   \if#1b\vbox
1591 <latexrelease>   \else\if #1t\top
1592 <latexrelease>   \else\ifmmode\center
1593 <latexrelease>   \else\@pboxswtrue $\center
1594 <latexrelease>   \fi\fi\fi
1595 <latexrelease>   \@parboxto{\let \hss \vss \let \unhbox \unvbox
1596 <latexrelease>     \csname bm@#3\endcsname}%
1597 <latexrelease>   \if@pboxsw \m@th $\fi
1598 <latexrelease>   \@end@tempboxa}
1599 <latexrelease>\plEndIncludeInRelease

```

`\underline` 下線を引く命令です。もとは `ltboxes.dtx` で定義されています。

```
1600 <latexrelease>\plIncludeInRelease{2019/10/01}{\underline}
1601 <latexrelease>                                {Make robust}%
1602 <*plcore | latexrelease>
1603 \DeclareRobustCommand\underline[1]{%
1604   \relax
1605   \ifmmode\@@underline{#1}%
1606   \else \leavevmode\null$\@@underline{\hbox{#1}}\m@th$\null\relax\fi}
1607 </plcore | latexrelease>
1608 <latexrelease>\plEndIncludeInRelease
1609 <latexrelease>\plIncludeInRelease{2016/04/17}{\underline}
1610 <latexrelease>                                {Remove extra \xkanjiskip}%
1611 <latexrelease>\def\underline#1{%
1612 <latexrelease>  \relax
1613 <latexrelease>  \ifmmode\@@underline{#1}%
1614 <latexrelease>  \else \leavevmode\null$\@@underline{\hbox{#1}}\m@th$\null\relax\fi}
1615 <latexrelease>\expandafter \let \csname underline \endcsname \@undefined
1616 <latexrelease>\plEndIncludeInRelease
1617 <latexrelease>\plIncludeInRelease{0000/00/00}{\underline}
1618 <latexrelease>                                {LaTeX2e original}%
1619 <latexrelease>\def\underline#1{%
1620 <latexrelease>  \relax
1621 <latexrelease>  \ifmmode\@@underline{#1}%
1622 <latexrelease>  \else $\@@underline{\hbox{#1}}\m@th$\relax\fi}
1623 <latexrelease>\expandafter \let \csname underline \endcsname \@undefined
1624 <latexrelease>\plEndIncludeInRelease
```

13 e-pTeX での FAM256 パッチの利用

`\e@alloc@chardef` L^AT_EX 2_ε 2015/01/01 以降、拡張レジスタがあれば利用するようになっていますの

`\e@alloc@top` で、e-pTeX の拡張レジスタを利用できるように設定します。

```
1625 <latexrelease>\plIncludeInRelease{2019/10/01}%
1626 <latexrelease>                                {\e@alloc@chardef}{Extended Allocation (FAM256)}%
1627 <*plcore | latexrelease>
1628 \ifx\widowpenalties\@undefined
```

オリジナルの T_EX の場合（拡張なしのアスキー pTeX の場合）。

```
1629   \mathchardef\e@alloc@top=255
1630   \let\e@alloc@chardef\chardef
1631 \else
1632   \ifx\omathchar\@undefined
```

e-T_EX 拡張で 2¹⁵ 個のレジスタが利用できます。

```
1633   \mathchardef\e@alloc@top=32767
1634   \let\e@alloc@chardef\mathchardef
1635 \else
```

FAM256 パッチが適用された e-pTeX の場合は、2¹⁶ 個のレジスタが利用できます。

```
1636   \omathchardef\e@alloc@top=65535
1637   \let\e@alloc@chardef\omathchardef
1638 \fi
1639 \fi
```

```

1640 </plcore | platexrelease>
1641 <platexrelease>\plEndIncludeInRelease
1642 <platexrelease>\plIncludeInRelease{2018/03/09}%
1643 <platexrelease>          {\e@alloc@chardef}{Extended Allocation (FAM256)}%
1644 <platexrelease>\ifx\omathchar\@undefined
1645 <platexrelease>  \ifx\widowpenalties\@undefined
1646 <platexrelease>    \mathchardef\e@alloc@top=255
1647 <platexrelease>    \let\e@alloc@chardef\chardef
1648 <platexrelease>  \else
1649 <platexrelease>    \mathchardef\e@alloc@top=32767
1650 <platexrelease>    \let\e@alloc@chardef\mathchardef
1651 <platexrelease>  \fi
1652 <platexrelease>\else
1653 <platexrelease>    \omathchardef\e@alloc@top=65535
1654 <platexrelease>    \let\e@alloc@chardef\omathchardef
1655 <platexrelease>\fi
1656 <platexrelease>\plEndIncludeInRelease
1657 <platexrelease>\plIncludeInRelease{2016/11/29}%
1658 <platexrelease>          {\e@alloc@chardef}{Extended Allocation (FAM256)}%
1659 <platexrelease>\ifx\omathchar\@undefined
1660 <platexrelease>  \ifx\widowpenalties\@undefined
1661 <platexrelease>    \mathchardef\e@alloc@top=255
1662 <platexrelease>    \let\e@alloc@chardef\chardef
1663 <platexrelease>  \else
1664 <platexrelease>    \mathchardef\e@alloc@top=32767
1665 <platexrelease>    \let\e@alloc@chardef\mathchardef
1666 <platexrelease>  \fi
1667 <platexrelease>\else
1668 <platexrelease>  \ifx\enablecjktoken\@undefined % pTeX
1669 <platexrelease>    \omathchardef\e@alloc@top=65535
1670 <platexrelease>    \let\e@alloc@chardef\omathchardef
1671 <platexrelease>  \else % upTeX
1672 <platexrelease>    \chardef\e@alloc@top=65535
1673 <platexrelease>    \let\e@alloc@chardef\chardef
1674 <platexrelease>  \fi
1675 <platexrelease>\fi
1676 <platexrelease>\plEndIncludeInRelease
1677 <platexrelease>\plIncludeInRelease{2015/01/01}%
1678 <platexrelease>          {\e@alloc@chardef}{LaTeX2e original}%
1679 <platexrelease>\ifx\widowpenalties\@undefined
1680 <platexrelease>  \mathchardef\e@alloc@top=255
1681 <platexrelease>  \let\e@alloc@chardef\chardef
1682 <platexrelease>\else
1683 <platexrelease>  \mathchardef\e@alloc@top=32767
1684 <platexrelease>  \let\e@alloc@chardef\mathchardef
1685 <platexrelease>\fi
1686 <platexrelease>\plEndIncludeInRelease
1687 <platexrelease>\plIncludeInRelease{0000/00/00}%
1688 <platexrelease>          {\e@alloc@chardef}{LaTeX2e original}%
1689 <platexrelease>\let\e@alloc@top\@undefined
1690 <platexrelease>\let\e@alloc@chardef\@undefined
1691 <platexrelease>\plEndIncludeInRelease

```

`\float@count` `\newcount` や `\newdimen` で使われます。

```
1692 < *plcore | latexrelease >
1693 \let\float@count\@alloc@top
1694 < /plcore | latexrelease >
```

`\e@mathgroup@top` 2015/01/01 以降の L^AT_EX 2_ε カーネルは、Xe_TE_X と Lua_TE_X に対して数式 fam の上限を 16 から 256 に増やしています (`\Umathcode` で判定)。FAM256 パッチが適用された e-p_TE_X でも同様に上限を 16 から 256 に増やします。これで

! LaTeX Error: Too many math alphabets used in version normal.

が出にくくなるはずです。

```
1695 < latexrelease > \plIncludeInRelease{2016/11/29}%
1696 < latexrelease > { \e@mathgroup@top } { Extended Allocation (FAM256) } %
1697 < *plcore | latexrelease >
1698 \ifx\omathchar\@undefined
1699 \chardef\e@mathgroup@top=16 % LaTeX2e kernel standard
1700 \else
1701 \mathchardef\e@mathgroup@top=256 % for e-pTeX FAM256 patched
1702 \fi
1703 < /plcore | latexrelease >
1704 < latexrelease > \plEndIncludeInRelease
1705 < latexrelease > \plIncludeInRelease{2015/01/01}%
1706 < latexrelease > { \e@mathgroup@top } { LaTeX2e original } %
1707 < latexrelease > \chardef\e@mathgroup@top=16
1708 < latexrelease > \plEndIncludeInRelease
1709 < latexrelease > \plIncludeInRelease{0000/00/00}%
1710 < latexrelease > { \e@mathgroup@top } { LaTeX2e original } %
1711 < latexrelease > \let\e@mathgroup@top\@undefined
1712 < latexrelease > \plEndIncludeInRelease
```

14 L^AT_EX 2_ε と pL^AT_EX 2_ε の更新タイミングずれ対策

`\l@nohyphenation` 通常は Babel のハイフネーション定義により提供されるパラメータです。しかし、L^AT_EX 2_ε 2017-04-15 以降・pL^AT_EX 2_ε 2017-04-08 以降では、`\verb` の途中でハイフネーションが起きないようにするために必須のものとなりました。L^AT_EX 2_ε は特殊な状況も想定して `ltfinal.dtx` で対策しているようですので、pL^AT_EX 2_ε も念のためここで対策します (参考: latex2e svn r1405)。

```
1713 < *plcore >
1714 \ifx\l@nohyphenation \@undefined
1715 \newlanguage\l@nohyphenation
1716 \fi
1717 < /plcore >
```

`\document@default@language` L^AT_EX 2_ε 2017-04-15 で導入されたパラメータですが、これに先立ち pL^AT_EX 2_ε 2017-04-08 でも使用しています。verbatim 環境の途中で改ページが起きた場合にヘッダでハイフネーションが抑制されないように、`\@outputpage` で `\language` をリセットするときに使われます (参考: latex2e svn r1407)。

```

1718 <latexrelease>\plIncludeInRelease{2017/04/08}{\document@default@language}%
1719 <latexrelease>                                {Save language for hyphenation}%
1720 <*/plcore | latexrelease>
1721 \ifx\document@default@language \@undefined
1722   \let\document@default@language\m@ne
1723 \fi
1724 </plcore | latexrelease>
1725 <latexrelease>\plEndIncludeInRelease
1726 <latexrelease>\plIncludeInRelease{0000/00/00}{\document@default@language}%
1727 <latexrelease>                                {Save language for hyphenation}%
1728 <latexrelease>\let\document@default@language\@undefined
1729 <latexrelease>\plEndIncludeInRelease

```

File e plext.dtx

15 概要

このパッケージは、以下の項目に関する機能を拡張するものです。

- 表組環境
- フロートとキャプションの出力位置
- 段落ボックス環境
- 作図環境
- 連数字、漢数字、傍点、下線
- 参照番号

このパッケージは縦組用クラス（`tarticle`, `tbook`, `treport`）のときには、自動的に読み込まれます。横組用クラス（`jarticle`, `jbook`, `jreport`）で拡張機能を使いたい場合は、文書ファイルのプリアンブルに以下の一行を記述してください。

```
\usepackage{plext}
```

16 組方向オプションについて

つぎの環境やコマンドは、組方向オプションが追加され、拡張されています。

- `tabular` 環境、`array` 環境
- `\layoutcaption` コマンド
- `minipage` 環境、`\parbox` コマンド、`\pbox` コマンド
- `picture` 環境

組方向オプションは、コマンド名や環境の後ろで`<と>`で囲って、“y”、“t”、“z”のいずれかを指定します。それぞれのオプションの意味はつぎのとおりです。デフォルトの組み方向は、横組のときは“y”、縦組のときは“t”です。

オプション	意味
y	横組で出力（横組モードでは何もしない）
t	縦組で出力（縦組モードでは何もしない）
z	90度回転して出力（横組モードでは何もしない）

組方向オプションを用いたサンプルを図 1 に示します。左から、“y”、“t”、“z” オプションを指定してあります。

たとえば、これはい たい何、いったいど うして、などと思え るようなことが世の 中にはたくさんあり ます。	たとえば、これはい たい何、いったいど うして、などと思え るようなことが世の 中にはたくさんあり ます？	たとえば、これはい たい何、いったいど うして、などと思え るようなことが世の 中にはたくさんあり ます！
--	--	--

Figure 1: 組方向オプションの使用例

17 コード

`\if@rotsw` このスイッチは、縦組モードで 90 度回転させるかどうかを示すのに使います。

```
1 <*package>
2 \newif\if@rotsw
```

17.1 表組環境

`tabular` 環境と `array` 環境は、組方向を指定するオプションを追加しました。これらのコマンドは、`l1tab.dtx` で定義されています。

`\array` `array` 環境と `tabular` 環境を開始するコマンドです。`tabular` 環境にはアスタリスク形式があります。

```
\tabular* 3 \def\array{\let\acol\@arrayacol \let\@classz\@arrayclassz
4 \let\@classiv\@arrayclassiv
5 \let\\\@arraycr\let\@halignto\@empty\X@tabarray}
6 %
7 \def\tabular{\let\@halignto\@empty\X@tabular}
8 \@namedef\tabular*{\@ifnextchar<%>
9 {\p@stabular}\p@stabular<Z>}}
```

`\X@tabarray` 組方向オプションを調べます。

```
\X@tabular 10 \def\X@tabarray{\@ifnextchar<%>
11 {\p@tabarray}\p@tabarray<Z>}}
12 \def\X@tabular{\@ifnextchar<%>
13 {\p@tabular}\p@tabular<Z>}}
```

`\p@stabular` アスタリスク形式の場合は、組方向オプションの後ろに幅を指定します。

```
\p@tabular 14 \def\p@stabular<#1>#2{%
15 \setlength\dimen@{#2}%
16 \edef\@halignto{to\the\dimen@}\p@tabular<#1>}
17 \def\p@tabular<#1>{\leavevmode \null\hbox \bgroup $\let\@acol\@tabacol
18 \let\@classz\@tabclassz
19 \let\@classiv\@tabclassiv \let\\\@tabularcr\p@tabarray<#1>}}
```

`\p@tabarray` 位置オプションを調べます。

```
20 \def\p@tabarray<#1>{\m@th\@ifnextchar[%]
21 {\p@array<#1>}\p@array<#1>[c]}}
```

`\p@array` tabular 環境と array 環境の内部形式です。

```
22 \def\p@array<#1>[#2]#3{%
23   \fork@array@option<#1>[#2]\@begin@alignbox
24   \bgroup\box@dir\adjustbaseline
25   \setbox\@arstrutbox\hbox{%
26     \iftdir
27       \if #1y\relax\yoko
28         \vrule\@height\arraystretch\ht\strutbox
29         \@depth\arraystretch\dp\strutbox \@width\z@
30       \else\if #1z\relax\@rotswtrue
31         \vrule\@height\arraystretch\ht\zstrutbox
32         \@depth\arraystretch\dp\zstrutbox \@width\z@
33       \else
34         \vrule\@height\arraystretch\ht\tstrutbox
35         \@depth\arraystretch\dp\tstrutbox \@width\z@
36       \fi\fi
37     \else
38       \if #1t\relax\tate
39         \vrule\@height\arraystretch\ht\tstrutbox
40         \@depth\arraystretch\dp\tstrutbox \@width\z@
41       \else
42         \vrule\@height\arraystretch\ht\strutbox
43         \@depth\arraystretch\dp\strutbox \@width\z@
44       \fi
45     \fi}%
46   \mkpream{#3}\edef\@preamble{\ialign \noexpand\@halignto
47     \bgroup \tabskip\z@skip \@arstrut \@preamble \tabskip\z@skip \cr}%
48   \let\@startpbox\@startpbox \let\@endpbox\@endpbox
49   \let\tabularnewline\\%

50   \let\par\@empty
51   \let\@sharp##%
52   \set@typeset@protect
53   \lineskip\z@skip\baselineskip\z@skip

54   \ifhmode \@preamerr\z@ \@@par\fi
55   \@preamble}
```

`\endarray` array 環境と tabular 環境の終了コマンドです。`\@end@alignbox` は `\p@array` から `\endtabular` 呼び出される `\fork@array@option` によって設定されます。

```
56 \def\endarray{\crr\egroup\egroup\@end@alignbox}
57 \def\endtabular{\crr\egroup\egroup\@end@alignbox $\egroup\null}
58 \expandafter \let \csname endtabular*\endcsname = \endtabular
```

`\fork@array@option` array 環境と tabular 環境で与えられた第一引数と第二引数の組合せの分岐を行いません。

コミュニティ版では、アスキー版で不自然だった表組 (array 環境および tabular 環境) と周囲の本文との揃え位置を修正し、以下のように設計しました。

- 周囲の組方向が横組かつ組方向が<y>, <z>指定の場合

- [t] 指定のとき
一行目のベースラインが周囲のそれと一致（罫線の場合は和文ベースラインの位置）
 - [c] 指定のとき
表組の中心が周囲の数式軸を通る（欧文ベースラインシフトの影響下）
 - [b] 指定のとき
最終行のベースラインが周囲のそれと一致（罫線の場合は和文ベースラインの位置）
- 周囲の組方向が横組かつ組方向が<t>指定の場合
 - [t] 指定のとき
表組の上端が周囲の和文ベースラインと一致
 - [c] 指定のとき
表組の中心が周囲の数式軸を通る（欧文ベースラインシフトの影響下）
 - [b] 指定のとき
表組の下端が周囲の和文ベースラインと一致
- 周囲の組方向が縦組かつ組方向が<y>指定の場合
 - [t] 指定のとき
表組の上端が周囲の和文ベースラインと一致
 - [c] 指定のとき
表組の中心が周囲の数式軸を通る（欧文ベースラインシフトの影響下）
 - [b] 指定のとき
表組の下端が周囲の和文ベースラインと一致
- 周囲の組方向が縦組かつ組方向が<t>指定の場合
 - [t] 指定のとき
一行目のベースラインが周囲のそれと一致（罫線の場合は和文ベースラインの位置）
 - [c] 指定のとき
表組の中心が周囲の数式軸を通る（欧文ベースラインシフトの影響下）
 - [b] 指定のとき
最終行のベースラインが周囲のそれと一致（罫線の場合は和文ベースラインの位置）
- 周囲の組方向が縦組かつ組方向が<z>指定の場合
 - [t] 指定のとき
一行目の欧文ベースラインが周囲のそれと一致

- [c] 指定のとき
表組の中心が周囲の数式軸を通る（欧文ベースラインシフトの影響下）
- [b] 指定のとき
最終行の欧文ベースラインが周囲のそれと一致

```

59 \def\fork@array@option<#1>[#2]{%
60 \@rotswfalse

縦組モードのとき：
61 \iftdir
62 \if #1y\relax\let\box@dir\yoko
63 \if #2t\relax
64 \def\@begin@alignbox{%
65 \@tempdima=\tbaselineshift
66 \advance\@tempdima-\ybaselineshift
67 \raise\@tempdima\top\bggroup\kern\z@\vtop}%
68 \let\@end@alignbox\egroup
69 \else\if #2b\relax
70 \def\@begin@alignbox{%
71 \@tempdima=\tbaselineshift
72 \advance\@tempdima-\ybaselineshift
73 \raise\@tempdima\box\bggroup\box}%
74 \def\@end@alignbox{\kern\z@\egroup}%
75 \else
76 \let\@begin@alignbox\center
77 \let\@end@alignbox\relax
78 \fi\fi
79 \else\if #1z\relax\let\box@dir\relax\@rotswtrue
80 \if #2t\relax
81 \def\@begin@alignbox{%
82 \@tempdima=\tbaselineshift
83 \advance\@tempdima-\ybaselineshift
84 \advance\@tempdima\ht\strutbox
85 \raise\arraystretch\@tempdima\top\bggroup\kern\z@\vtop}%
86 \let\@end@alignbox\egroup
87 \else\if #2b\relax
88 \def\@begin@alignbox{%
89 \@tempdima=\tbaselineshift
90 \advance\@tempdima-\ybaselineshift
91 \advance\@tempdima-\dp\strutbox
92 \raise\arraystretch\@tempdima\box\bggroup\box}%
93 \def\@end@alignbox{\kern\z@\egroup}%
94 \else
95 \let\@begin@alignbox\center
96 \let\@end@alignbox\relax
97 \fi\fi
98 \else\let\box@dir\tate
99 \if #2t\relax
100 \let\@begin@alignbox\top
101 \let\@end@alignbox\relax
102 \else\if #2b\relax
103 \let\@begin@alignbox\box
104 \let\@end@alignbox\relax

```

```

105 \else
106 \let\@begin@alignbox\vcenter
107 \let\@end@alignbox\relax
108 \fi\fi
109 \fi\fi

横組モードのとき：

110 \else
111 \if #1t\relax\let\box@dir\tate
112 \if #2t\relax
113 \def\@begin@alignbox{\vtop\bgroup\kern\z@\vbox}%
114 \let\@end@alignbox\egroup
115 \else\if #2b\relax
116 \def\@begin@alignbox{\vbox\bgroup\vbox}%
117 \def\@end@alignbox{\kern\z@\egroup}%
118 \else
119 \let\@begin@alignbox\vcenter
120 \let\@end@alignbox\relax
121 \fi\fi
122 \else\let\box@dir\yoko
123 \if #2t\relax
124 \let\@begin@alignbox\vtop
125 \let\@end@alignbox\relax
126 \else\if #2b\relax
127 \let\@begin@alignbox\vbox
128 \let\@end@alignbox\relax
129 \else
130 \let\@begin@alignbox\vcenter
131 \let\@end@alignbox\relax
132 \fi\fi
133 \fi\fi}

```

17.2 フロートとキャプションの出力位置

キャプションとフロートは、出力位置の指定や大きさの指定などができるように拡張しています。詳細は、『日本語 L^AT_EX 2_ε ブック』を参照してください。

`\layoutfloat` コマンドで作られるボックスです。

```
134 \newbox\@floatbox
```

フロートオブジェクトの幅と高さです。

```
135 \newdimen\floatwidth
```

```
136 \newdimen\floatheight
```

フロートオブジェクトのまわりに引かれる罫線の太さです。

```
137 \newdimen\floatruletick \floatruletick=0.4pt
```

フロートオブジェクトとキャプションの間のアキです。

```
138 \newdimen\captionfloatsep \captionfloatsep=10pt
```

`\caption@dir` には、キャプションを組む方向を示すオプションが格納されます。

`\captiondir` は `\caption@dir` の値と現在の組み方向によって、`\yoko`, `\tate`, `\relax` のいずれかに設定されます。

```

139 \def\caption@dir{Z}
140 \let\captiondir\relax

    キャプションの幅です。
141 \newdimen\captionwidth \captionwidth\z@

    キャプションを付ける位置を指定します。
142 \def\caption@posa{Z}
143 \def\caption@posb{Z}

    組み立てられたキャプションが格納されるボックスです。
144 \newbox\@captionbox

    キャプションに使われる文字です。
145 \def\captionfontsetup{\normalfont\normalsize}

```

`\layoutfloat` `\layoutfloat` は図表類の大きさと位置を指定するのに使います。大きさを省略するか、負の値を指定すると、そのオブジェクトの自然な長さになります。このとき `\@layoutfloat` は、罫が引かれません。正の大きさを指定すると、`\floatruletick` の太さの罫で囲まれます。

位置指定を省略した場合、中央揃えになるようにしています。

```

146 \def\layoutfloat{\@ifnextchar(%)
147   {\X@layoutfloat}{\X@layoutfloat(-5\p@,-5\p@)}}
148 %
149 \def\X@layoutfloat(#1,#2){\@ifnextchar[%]
150   {\@layoutfloat(#1,#2)}{\@layoutfloat(#1,#2)[c]}}
151 %
152 \long\def\@layoutfloat(#1,#2)[#3]#4{%
153   \setbox\z@\hbox{#4}%
154   \floatwidth=#1 \floatheight=#2 \edef\float@pos{#3}%
155   \ifdim\floatwidth<\z@
156     \floatwidth\wd\z@\floatruletick\z@
157   \fi
158   \ifdim\floatheight<\z@
159     \floatheight\ht\z@\advance\floatheight\dp\z@\relax
160     \floatruletick\z@
161   \fi
162   \setbox\@floatbox\vbox to\floatheight{\offinterlineskip
163     \hrule width\floatwidth height\floatruletick depth\z@
164     \vss\hbox to\floatwidth{%
165       \vrule width\floatruletick height\floatheight depth\z@
166       \hss\vbox to\floatheight{\hsize\floatwidth\vss#4\vss}\hss
167       \vrule width\floatruletick height\floatheight depth\z@
168     }\hrule width\floatwidth height\floatruletick depth\z@}}

```

`\DeclareLayoutCaption` `\DeclareLayoutCaption` コマンドは、キャプションの組方向、付ける位置や幅のデフォルトをフロートのタイプごとに設定することができます。このコマンドでデフォルト値が設定されていないと、`\pcaption` コマンドでエラーが発せられます。このコマンドはブリアンブルでのみ、使用できます。

```
\DeclareLayoutCaption<type><dir>>(<width>)[<pos1>]<pos2>]
```

コマンド引数を省略することはできません。 $\langle dir \rangle$ には、‘y’、‘t’、‘z’、‘n’のいずれかを指定します。‘n’と指定をすると、本文の組み方向と同じ方向でキャプションが組まれます。これがデフォルトです。(補足：2018/09/20 v1.2j までは、この説明に反して実際のコードは‘y’がデフォルトになっており、「日本語 L^AT_EX 2_ε ブック」にも‘y’と書かれていましたが、後述の★のバグ修正に合わせ、2018/10/07 v1.2k で‘n’に直しました。)

$\langle width \rangle$ には、キャプションを折り返す長さを指定します。‘(12zw)’と指定をすると、漢字12文字分の長さで折り返されます。デフォルトは $(.8\text{\linewidth})$ です。なお、キャプションの幅をフロートオブジェクトの幅に合わせる場合は‘ (\floatwidth) ’と指定し、高さに合わせる場合は‘ (\floatheight) ’と指定します。

$\langle pos1 \rangle$ と $\langle pos2 \rangle$ には、キャプションを出力する位置を指定します。 $\langle pos1 \rangle$ は、‘c’、‘t’、‘b’のいずれかです。 $\langle pos2 \rangle$ は、‘u’、‘d’、‘l’、‘r’のいずれかです。デフォルトは、figure タイプが‘cd’、table タイプは‘cu’です。

```

169 \def\DeclareLayoutCaption#1<#2>(<#3>)[<#4#5>]{%
170   \expandafter
171   \ifx\csname #1@layoutc@ption\endcsname\relax \else
172     \@latex@info{Redeclaring capiton layout setting of ' #1' }%
173   \fi
174   \expandafter
175   \gdef\csname #1@layoutc@ption\endcsname{%
176     \if Z\caption@dir\def\caption@dir{#2}\fi
177     \ifdim\captionwidth=\z@ \captionwidth=#3\relax\fi
178     \if Z\caption@posa\def\caption@posa{#4}\fi
179     \if Z\caption@posb\def\caption@posb{#5}\fi}}
180 \@onlypreamble\DeclareLayoutCaption

181 \DeclareLayoutCaption{figure}<n>(.8\linewidth)[cd]
182 \DeclareLayoutCaption{table}<n>(.8\linewidth)[cu]

```

$\backslash layoutcaption$ $\backslash DeclareLayoutCaption$ コマンドで設定をした、デフォルト値とは異なる設定で $\backslash X@layoutcaption$ 組みたい場合は、 $\backslash layoutcaption$ コマンドを使用します。

$\backslash @ilayoutcaption$ $\backslash layoutcaption<\langle dir \rangle>(<\langle width \rangle>)[<\langle pos \rangle]$

$\backslash @iilayoutcaption$ なお、 $\backslash layoutcaption$ に組み方向オプションを付けましたので、 $\backslash captiondir$ で組み方向を指定する必要はありません。また、 $\backslash captiondir$ で指定をしても、その値は無視されます。

```

183 \def\layoutcaption{\def\caption@dir{Z}\captionwidth\z@
184   \def\caption@posa{Z}\def\caption@posb{Z}%
185   \@ifnextchar<\X@layoutcaption{%
186     \@ifnextchar(\@ilayoutcaption{%
187       \@ifnextchar[\@iilayoutcaption\relax]}}
188 %
189 \def\X@layoutcaption<#1>{\def\caption@dir{#1}%
190   \@ifnextchar(\@ilayoutcaption{%
191     \@ifnextchar[\@iilayoutcaption\relax]}
192 %
193 \def\@ilayoutcaption(#1){\setlength\captionwidth{#1}%
194   \@ifnextchar[\@iilayoutcaption]{\relax}}
195 %

```

```

196 \def\@iilayoutcaption[#1#2]{%
197   \def\caption@posa{#1}\def\caption@posb{#2}}

```

`\pccaption` キャプションを図表類の天地左右の指定箇所に付けるには `\pccaption` コマンドで指定 `\@pccaption` をします。位置の指定は `\layoutcaption` コマンドで行ないます。`\layoutcaption` コマンドが省略された場合は、`\DeclareLayoutCaption` コマンドで設定されているデフォルト値が使われます。

```

198 \def\pccaption{%
199   \ifx\@cctype\undefined
200     \@latex@error{\noexpand\pccaption outside float}\@ehd
201     \expandafter\@gobble
202   \else
203     \refstepcounter\@cctype
204     \expandafter\@firstofone
205   \fi
206   {\@dblarg{\@pccaption\@cctype}}}%
207 }
208 %
209 \long\def\@pccaption#1[#2]#3{%
210   \addcontentsline{\csname ext#1\endcsname}{#1}{%
211     \protect\numberline{\csname the#1\endcsname}{\ignorespaces#2}}%
212   \ifvoid\@floatbox
213     \@latex@error{Use \noexpand\pccaption with ‘\string\layoutfloat’}\@eha
214   \fi
215   \make@pccaptionbox{#3}%
216   \@pboxswfalse
217   \setbox\@tempboxa\vbox{\hbox to\hsize{\if l\float@pos\else\hss\fi
218     \if l\caption@posb\box\@captionbox\kern\captionfloatsep\fi
219     \if t\caption@posa\vtop
220       \else\if b\caption@posa\vbox
221         \else\@pboxswtrue $\vcenter \fi\fi
222     {\if u\caption@posb\box\@captionbox\kern\captionfloatsep\fi
223       \unvbox\@floatbox
224       \if d\caption@posb\kern\captionfloatsep\box\@captionbox\fi}%
225     \if@pboxsw \m@th$\fi
226     \if r\caption@posb\kern\captionfloatsep\box\@captionbox\fi
227     \if r\float@pos\else\hss\fi}}}%
228   \par\vskip.25\baselineskip
229   \box\@tempboxa}

```

`\make@pccaptionbox` キャプションを組み立て、`\@captionbox` を作成します。

```

230 \def\make@pccaptionbox#1{%

```

まず、デフォルトの設定がされているかを確認します。設定されていない場合は、警告メッセージを出力し、現在の組モードでのデフォルト値を使用します。設定されていれば、そのデフォルト値にします。

```

231   \expandafter
232   \ifx\csname\@cctype @layoutc@ption\endcsname\relax
233     \@latex@warning{Default caption layout of ‘\@cctype’ unknown}%
234     \def\caption@dir{Z}\captionwidth\z@
235     \def\caption@posa{Z}\def\caption@posb{Z}%
236   \else

```

```

237 \csname \@captype @layoutc@ption\endcsname
238 \fi

```

次に、組み方向を設定します。基本組の組み方向とキャプションの組み方向を変える場合には、`\@tempswa` を真とします。文字を回転させるときは `\@rotsw` を真にします。

```

239 \@rotswfalse \@tempswafalse
240 \iftdir\if y\caption@dir \let\captiondir\yoko \@tempswatrue
241 \else\if z\caption@dir \let\captiondir\relax \@rotswtrue
242 \else\let\captiondir\tate\fi\fi
243 \else\if t\caption@dir\let\captiondir\tate \@tempswatrue
244 \else\let\captiondir\yoko\fi
245 \fi

```

キャプションを組み立てる前に、まず、キャプション文字列がどの程度の長さを持っているのかを確認するために、`\hbox` に入れます。

```

246 \setbox0\hbox{\if@rotsw $\fi\hbox{\captiondir
247 \captionfontsetup\parindent\z@\inhibitglue
248 \csname fnum@\@captype\endcsname\char\@euc"A1A1\relax#1}%
249 \if@rotsw \m@th$\fi}%

```

キャプションの幅に合わせるため、再び、ボックスを組み立てます。

キャプションを折り返さなくてもよい場合、`\@tempdima` をキャプションの長さにします。ただし、キャプションの組み方向が基本組の組み方向と異なる場合（`\@tempswa` が真）は、ボックス 0 の幅ではなく、高さに設定をします（ pTeX では同じボックスでも、組方向によって `\wd` と `\ht+\wd` の返す寸法が異なることに注意）。`\captionwidth` の値が、キャプションの幅よりも長い場合、折り返さなくてはなりませんので、`\@tempdima` を `\captionwidth` にします。

日本語 TeX 開発コミュニティによる修正：2018/09/20 v1.2j までは、キャプションの組方向が基本組の組み方向と直交する場合に深さを考慮するのを忘れていたために、本来は折り返さずに済むはずの短いキャプションが、必ず折り返されてしまうというバグ（★）がありました。2018/10/07 v1.2k でこのバグを修正したため、組版結果が大きく変わる場合があります。

```

250 \if@tempswa \@tempdima\ht0 \advance\@tempdima\dp0
251 \else \@tempdima\wd0 \fi
252 \ifdim\@tempdima>\captionwidth \@tempdima\captionwidth \fi
253 \@pboxswfalse
254 \setbox0\hbox{\if@rotsw $\fi
255 \if u\caption@posb\vbox
256 \else\if d\caption@posb\vbox
257 \else\if t\caption@posa\top
258 \else\if b\caption@posa\vbox
259 \else\ifmmode\vcenter\else\@pboxswtrue $\vcenter\fi
260 \fi\fi\fi
261 {\hsize\@tempdima\kern\z@
262 \vbox{\captiondir\hsize\@tempdima
263 \captionfontsetup\parindent\z@\inhibitglue
264 \csname fnum@\@captype\endcsname\char\@euc"A1A1\relax#1}\kern\z@
265 }\if@pboxsw \m@th$\fi \if@rotsw \m@th$\fi}%

```

最後に `\@captionbox` を組み立てます。

位置 2 オプションが 'u' か 'd' の場合、このボックスの幅をフロートオブジェクトの幅と同じ長さにし、位置 1 オプションでの揃えに組み立てます。

位置 2 オプションが 'l' か 'r' の場合は、キャプションの幅です。このときの位置 1 オプションの揃えは、この前の段階で準備をしておき、`\@pcaption` で最終的にフロートオブジェクトと組み合わせるときになされます。

```
266 \let\to@captionboxwidth\relax
267 \if l\caption@posb \else\if r\caption@posb\else
268 \def\to@captionboxwidth{to\floatwidth}\fi\fi
269 \setbox\@captionbox\hbox\to@captionboxwidth{%
270   \if t\caption@posa\else\hss\fi
271   \unhbox0\relax
272   \if b\caption@posa\else\hss\fi}}
```

17.3 段落ボックス環境

`minipage` 環境と `\parbox` コマンドも、`tabular` 環境と同じように、組方向を指定するオプションを追加してあります。これらのコマンドは、`ltboxes.dtx` で定義されています。

`\parbox` コマンドは幅だけでなく高さも指定できるようになっています。新しい `\parbox` コマンドについての詳細は、`usrguide.tex` を参照してください。

`minipage` 環境

`\minipage` 組方向オプションを調べます。

```
273 \def\minipage{\@ifnextchar<%>
274   {\X@minipage}{\X@minipage<Z>}}
```

`\X@minipage` 位置オプションを調べます。

```
275 \def\X@minipage<#1>{\@ifnextchar[%]
276   {\@iminipage<#1>}{\@iiminipage<#1>{c}\relax[s]}}
```

`\@iminipage` 高さオプションを調べます。

```
277 \def\@iminipage<#1>[#2]{\@ifnextchar[%]
278   {\@iiminipage<#1>{#2}}{\@iiminipage<#1>{#2}\relax[s]}}
```

`\@iiminipage` 内部位置オプションを調べます。

```
279 \def\@iiminipage<#1>#2[#3]{\@ifnextchar[%]
280   {\@iiiminipage<#1>{#2}{#3}}{\@iiiminipage<#1>{#2}{#3}[#2]}}
```

`\@iiiminipage` `minipage` 環境の内部形式です。`\leavevmode` の後の `\bgroup` は、回転オプションが指定されたときのフラグ `\if@rotsw` が、このマクロの内部だけで有効になるようにするためです。この括弧は、`\endminipage` コマンドで閉じます。

```
281 \def\@iiiminipage<#1>#2#3[#4]#5{%
282   \leavevmode\bgroup
283   \setlength\@tempdima{#5}%
284   \def\@mpargs{<#1>{#2}{#3}[#4]{#5}}%
```



```

285 \@rotswfalse
286 \iftkdir
287   \if #1y\relax\let\box@dir\yoko
288   \else\if #1z\relax\@rotswtrue \let\box@dir\relax
289   \else\let\box@dir\tate
290   \fi\fi
291 \else
292   \if #1t\relax\let\box@dir\tate
293   \else\let\box@dir\yoko
294   \fi
295 \fi
296 \setbox\@tempboxa\vbox\bgroup\box@dir
297   \if@rotsw \hsize\@tempdima\hbox\bgroup$\vbox\bgroup\fi

298 \adjustbaseline
299 \color@begingroup
300   \hsize\@tempdima
301   \textwidth\hsize \columnwidth\hsize
302   \@parboxrestore
303   \def\@mpfn{mpfootnote}\def\thempfn{\thempfootnote}%
304   \c@mpfootnote\z@
305   \let\@footnotetext\@mpfootnotetext
306   \let\@listdepth\@mplistdepth \@mplistdepth\z@
307   \@minipagerestore
308   \@setminipage}

```

`\endminipage` minipage 環境の終了コマンドです。

```

309 \def\endminipage{%
310   \par
311   \unskip
312   \ifvoid\@mpfootins\else
313     \vskip\skip\@mpfootins
314     \normalcolor
315     \footnoterule
316     \unvbox\@mpfootins
317   \fi
318   \@minipagefalse   %% added 24 May 89
319   \color@endgroup
320   \if@rotsw \egroup\m@th$\egroup\fi

  \@iiminipage で開始したグループを閉じるための \egroup です。
321   \egroup
322   \expandafter\@iiparbox\@mpargs{\unvbox\@tempboxa}\egroup}

```

`\parbox` コマンド

`\parbox` 組方向オプションを調べます。

```

323 \DeclareRobustCommand\parbox{\@ifnextchar<%>
324   {\X@parbox}{\X@parbox<Z>}}

```

`\X@parbox` 位置オプションを調べます。

```

325 \def\X@parbox<#1>{\@ifnextchar[%]
326   {\@iparbox<#1>}{\@iiparbox<#1>{c}\relax[s]}}

```

`\@iparbox` 高さオプションを調べます。

```
327 \def\@iparbox<#1>[#2]{\@ifnextchar[%]  
328   {\@iiiparbox<#1>[#2]}\@iiiparbox<#1>[#2]\relax[s]}}
```

`\@iiparbox` 内部位置オプションを調べます。

```
329 \def\@iiparbox<#1>#2[#3]{\@ifnextchar[%]  
330   {\@iiiiparbox<#1>[#2]#3}\@iiiparbox<#1>[#2]#3[#2]}}
```

`\@iiiiparbox` `parbox` の内部形式です。 `minipage` 環境と同じようにグルーピングをします。この括弧と対になるのは、このマクロの最後の `\egroup` です。

```
331 \long\def\@iiiiparbox<#1>#2#3[#4]#5#6{%  
332   \leavevmode\null\bgroup  
333   \setlength\@tempdima{#5}%  
334   \fork@parbox@option<#1>[#2]%  
335   \if@rotsw  
336     \@begin@tempboxa\vbox{\box@dir\hsize\@tempdima  
337       \hbox{$\vbox{\@parboxrestore\adjustbaseline#6\@par}\m@th$}}%  
338   \else  
339     \@begin@tempboxa\vbox{\box@dir  
340       \hsize\@tempdima\@parboxrestore\adjustbaseline#6\@par}%  
341   \fi  
342   \ifx\relax#3\relax\else  
343     \setlength\@tempdimb{#3}%  
344     \edef\@parboxto{to\the\@tempdimb}%  
345   \fi  
346   \@begin@parbox\@parboxto{\box@dir\adjustbaseline  
347     \let\hss\vss\let\unhbox\unvbox  
348     \csname bm@#4\endcsname}\@end@parbox  
349   \@end@tempboxa\egroup\null}
```

`\fork@parbox@option` `\parbox` で与えられた第一引数と第二引数の組合せの分岐を行ないます。

コミュニティ版では、アスキー版で不自然だった `\parbox` の箱と周囲の本文との揃え位置を修正し、以下のように設計しました。

- 周囲の組方向が横組かつ組方向が<y>, <z>指定の場合
 - － [t] 指定のとき
一行目のベースラインが周囲のそれと一致
 - － [c] 指定のとき
箱の中心が周囲の数式軸を通る（欧文ベースラインシフトの影響下）
 - － [b] 指定のとき
最終行のベースラインが周囲のそれと一致
- 周囲の組方向が横組かつ組方向が<t>指定の場合
 - － [t] 指定のとき
箱の上端が周囲の和文文字の高さと一致

- [c] 指定のとき
箱の中心が周囲の数式軸を通る（欧文ベースラインシフトの影響下）
 - [b] 指定のとき
箱の下端が周囲の和文文字の深さと一致
- 周囲の組方向が縦組かつ組方向が<y>指定の場合
 - [t] 指定のとき
箱の上端が周囲の和文文字の高さと一致
 - [c] 指定のとき
箱の中心が周囲の数式軸を通る（欧文ベースラインシフトの影響下）
 - [b] 指定のとき
箱の下端が周囲の和文文字の深さと一致
- 周囲の組方向が縦組かつ組方向が<t>指定の場合
 - [t] 指定のとき
一行目のベースラインが周囲のそれと一致
 - [c] 指定のとき
箱の中心が周囲の数式軸を通る（欧文ベースラインシフトの影響下）
 - [b] 指定のとき
最終行のベースラインが周囲のそれと一致
- 周囲の組方向が縦組かつ組方向が<z>指定の場合
 - [t] 指定のとき
箱の上端が周囲の和文文字の高さと一致
 - [c] 指定のとき
箱の中心が周囲の数式軸を通る（欧文ベースラインシフトの影響下）
 - [b] 指定のとき
箱の下端が周囲の和文文字の深さと一致

```
350 \def\fork@parbox@option<#1>[#2]{%
351 \@rotswfalse
```

縦組モードのとき：

```
352 \iftdir
353 \if #1y\relax\let\box@dir\yoko
354   \if #2t\relax
355     \def\@begin@parbox{\raise\cht\vtop\bgroup\kern\z@\vtop}%
356     \let\@end@parbox\egroup
357   \else\if #2b\relax
358     \def\@begin@parbox{\lower\cdp\vbox\bgroup\vbox}%
359     \def\@end@parbox{\kern\z@\egroup}%
360   \else\ifmmode
```

```

361     \let\@begin@parbox\vcenter
362     \let\@end@parbox\relax
363   \else
364     \def\@begin@parbox{\$ \vcenter}%
365     \def\@end@parbox{\m@th}%
366   \fi\fi\fi
367 \else\if #1z\relax\@rotswtrue \let\box@dir\relax
368   \if #2t\relax
369     \def\@begin@parbox{\raise\cht\vtop\bgroup\kern\z@\vtop}%
370     \let\@end@parbox\egroup
371   \else\if #2b\relax
372     \def\@begin@parbox{\lower\cdp\vbox\bgroup\vbox}%
373     \def\@end@parbox{\kern\z@\egroup}%
374   \else\ifmmode
375     \let\@begin@parbox\vcenter
376     \let\@end@parbox\relax
377   \else
378     \def\@begin@parbox{\$ \vcenter}%
379     \def\@end@parbox{\m@th}%
380   \fi\fi\fi
381 \else\let\box@dir\tate
382   \if #2t\relax
383     \let\@begin@parbox\vtop
384     \let\@end@parbox\relax
385   \else\if #2b\relax
386     \let\@begin@parbox\vbox
387     \let\@end@parbox\relax
388   \else\ifmmode
389     \let\@begin@parbox\vcenter
390     \let\@end@parbox\relax
391   \else
392     \def\@begin@parbox{\$ \vcenter}%
393     \def\@end@parbox{\m@th}%
394   \fi\fi\fi
395 \fi\fi

```

横組モードのとき：

```

396 \else
397 \if #1t\relax\let\box@dir\tate
398   \if #2t\relax
399     \def\@begin@parbox{\raise\cht\vtop\bgroup\kern\z@\vtop}%
400     \let\@end@parbox\egroup
401   \else\if #2b\relax
402     \def\@begin@parbox{\lower\cdp\vbox\bgroup\vbox}%
403     \def\@end@parbox{\kern\z@\egroup}%
404   \else\ifmmode
405     \let\@begin@parbox\vcenter
406     \let\@end@parbox\relax
407   \else
408     \def\@begin@parbox{\$ \vcenter}%
409     \def\@end@parbox{\m@th}%
410   \fi\fi\fi
411 \else\let\box@dir\yoko
412   \if #2t\relax

```

```

413 \let\@begin@parbox\vtop
414 \let\@end@parbox\relax
415 \else\if #2b\relax
416 \let\@begin@parbox\vbox
417 \let\@end@parbox\relax
418 \else\ifmmode
419 \let\@begin@parbox\vcenter
420 \let\@end@parbox\relax
421 \else
422 \def\@begin@parbox{$\vcenter}%
423 \def\@end@parbox{\m@th$}%
424 \fi\fi\fi
425 \fi\fi}

```

\pbox コマンド

\pbox は組み方向を指定できるボックスコマンドです。次のような構文となっています。

```
\pbox<dir>[<width>][<pos>]{<obj>}
```

\pbox オプションを調べます。

```
426 \DeclareRobustCommand\pbox{\leavevmode\@ifnextchar<{\X@makePbox}{\X@makePbox<Z>}}
```

\X@makepbox

```

\@imakepbox 427 \def\X@makePbox<#1>{%
428 \@ifnextchar[{\@imakePbox<#1>}{\@imakePbox<#1>[-5\p@]}}
429 %
430 \def\@imakePbox<#1>[#2]{\@ifnextchar[%]
431 {\@iimakePbox<#1>[#2]}{\@iimakePbox<#1>[#2][c]}}
```

\@iimakePbox \pbox の内部形式です。

```

432 \def\@iimakePbox<#1>#2[#3]#4{%
433 \bgroup \@rotswfalse \@pboxswfalse
434 \iftdir
435 \if #1y\relax\let\box@dir\yoko
436 \else\if #1z\relax\@rotswtrue \let\box@dir\relax
437 \else\let\box@dir\tate
438 \fi\fi
439 \else
440 \if #1t\relax\let\box@dir\tate
441 \else\let\box@dir\yoko
442 \fi
443 \fi
444 \ifmmode\else\if@rotsw\@pboxswtrue\hbox\bgroup$\fi\fi
445 \setlength{\@tempdima}{#2}%
446 \ifdim\@tempdima<\z@ \hbox{\box@dir#4}\else
447 \hb@xt@\@tempdima{\box@dir
448 \if #3l\relax\else\hss\fi
449 #4\relax
450 \if #3r\relax\else\hss\fi}\fi
451 \if@pboxsw \m@th$\egroup\fi\egroup}

```

17.4 作図環境

picture 環境も、組方向を指定するオプションを追加してあります。なお、これらのコマンドは、`ltpictur.dtx` で定義されています。

`\picture` 組方向オプションを調べます。

```
452 \def\picture{\@ifnextchar<%>
453   {\X@picture}{\X@picture<Z>}}
```

`\X@picture` 図形領域オプションを調べます。

```
454 \def\X@picture<#1>(#2,#3){\@ifnextchar(%)
455   {\@@picture<#1>(#2,#3)}{\@@picture<#1>(#2,#3)(0,0)}}
```

`\@@picture` picture 環境の内部ではベースラインシフトの値をゼロにします。以前に設定されていた値は、それぞれ保存され、終了時に、その値に戻されます。

```
456 \newdimen\save@ybaselineshift
457 \newdimen\save@tbaselineshift
458 \newdimen\@picwd
```

`\X@picture@dimens` 図形領域の寸法値を調整する命令を切り出しました。基本組の組み方向と直交する場合は、高さと幅を入れ替えます。

```
459 \ifx\@defaultunitsset\undefined      % old
460   \def\X@picture@dimens#1#2#3#4{%
461     \@picwd=#1\unitlength \@picht=#2\unitlength
462     \@tempdima=#3\unitlength \@tempdimb=#4\unitlength
463   }
464 \else                                  % 2020-10-01
465   \def\X@picture@dimens#1#2#3#4{%
466     \@defaultunitsset\@picwd{#1}\unitlength
467     \@defaultunitsset\@picht{#2}\unitlength
468     \@defaultunitsset\@tempdima{#3}\unitlength
469     \@defaultunitsset\@tempdimb{#4}\unitlength
470   }
471 \fi
```

`\picture` の内部形式です。3 組目の引数は、原点座標です。

```
472 \def\@@picture<#1>(#2,#3)(#4,#5){%
473   \save@ybaselineshift\ybaselineshift
474   \save@tbaselineshift\tbaselineshift
475   \iftdir
476     \if#1y\let\box@dir\yoko
477     \X@picture@dimens{#3}{#2}{#5}{#4}%
478   \else\let\box@dir\tate
479     \X@picture@dimens{#2}{#3}{#4}{#5}%
480   \fi
481 \else
482   \if#1t\let\box@dir\tate
483     \X@picture@dimens{#3}{#2}{#5}{#4}%
484   \else\let\box@dir\yoko
485     \X@picture@dimens{#2}{#3}{#4}{#5}%
486   \fi
487 \fi
```

```

488 \setbox\@picbox\hb@xt@\@picwd\bgroup\box@dir
489 \hskip-\@tempdima
490 \lower\@tempdimb\hbox\bgroup
491 \ybaselineshift\z@ \tbaselineshift\z@
492 \ignorespaces}

```

`\endpicture` 図形領域の幅と高さを指定の大きさにしてから、出力をします。そして、最後にベースラインシフトの値を元に戻します。

```

493 \def\endpicture{%
494 \egroup\hss\egroup
495 \ht\@picbox\@picht \wd\@picbox\@picwd \dp\@picbox\z@
496 \mbox{\box\@picbox}%
497 \ybaselineshift\save@ybaselineshift
498 \tbaselineshift\save@tbaselineshift}

```

`\put` `picture` 環境の内部で、フォントサイズ変更コマンドなどが使用された場合、ベースラインシフト量が新たに設定されてしまうため、これらのコマンドがベースラインシフトの影響を受けないように再定義をします。ベースラインシフトを有効にした場合は、`\pbox` コマンドを使用してください。

```

\oval 499 \let\org@put\put
\circle 500 \def\put{\ybaselineshift\z@\tbaselineshift\z@\org@put}
501 %
502 \let\org@line\line
503 \def\line{\ybaselineshift\z@\tbaselineshift\z@\org@line}
504 %
505 \let\org@vector\vector
506 \def\vector{\ybaselineshift\z@\tbaselineshift\z@\org@vector}
507 %
508 \let\org@dashbox\dashbox
509 \def\dashbox{\ybaselineshift\z@\tbaselineshift\z@\org@dashbox}
510 %
511 \let\org@oval\oval
512 \def\oval{\ybaselineshift\z@\tbaselineshift\z@\org@oval}
513 %
514 \let\org@circle\circle
515 \def\circle{\ybaselineshift\z@\tbaselineshift\z@\org@circle}

```

17.5 連数字／漢数字／傍点／下線

ここでは、連数字、漢数字、傍点、下線について説明をしています。

連数字と漢数字、および傍点と下線についての詳細は、『日本語 L^AT_EX 2_ε ブック』を参照してください。なお、傍点に使う文字は `pldefs.ltx` で定義されています。

なお、連数字コマンドは3種類ありましたが、`\rensuji` コマンド一つにまとめました。新しい連数字コマンドは次の構文となります。

```

\rensuji[⟨pos⟩]⟨横に並べる半角文字⟩
\rensuji* [⟨pos⟩]⟨横に並べる半角文字⟩

```

アスタリスク形式の場合は、行間を連数字の幅に合わせて広げません。⟨pos⟩は、連数字を揃える位置です。‘c’（中央揃え）、‘r’（右寄せ）、‘l’（左寄せ）を指定でき

ます。デフォルトでは、中央に揃えます。

次のフラグが真の場合には、連数字の幅に合わせて行間を広げません。アスタリスク形式の場合に真になります。

```
516 \newif\ifnot@advanceline
```

`\rensuji` は連数字の前後に入るアキです。デフォルトは、現在の文字の幅の 4 分の 1 を基準にしています。

```
517 \newskip\rensuji
```

```
518 \rensuji=0.25\ch plus.25zw minus.25zw
```

連数字

`\rensuji` は、*形式かどうかを調べます。`\@rensuji` は、位置オプションを調べます。`\@@rensuji` が `\rensuji` の内部形式です。

```
\@@rensuji 519 \DeclareRobustCommand\rensuji{%
520   \@ifstar{\not@advancelinetrue\@rensuji}{\@rensuji}
521   \def\@rensuji{\@ifnextchar[{\@rensuji}{\@rensuji[c]}}
522   \def\@@rensuji[#1]#2{%
523     \ifvmode\leavevmode\fi
524     \iftdir\ifmdir\hbox{#2}\else
525       \hskip\rensuji
526       \ifnot@advanceline\not@advancelinefalse\else
527         \setbox\z@\hbox{\yoko#2}%
528         \@tempdima\ht\z@ \advance\@tempdima\dp\z@
529         \if #1c\relax\vrule\@width\z@ \@height.5\@tempdima \@depth.5\@tempdima
530         \else\if #1r\relax\vrule\@width\z@\@height\z@ \@depth\@tempdima
531         \else\vrule\@width\z@ \@height\@tempdima \@depth\z@
532         \fi\fi
533       \fi
534       \if #1c\relax\hbox to1zw{\yoko\hss#2\hss}%
535       \else\if #1r\relax\vbbox{\hbox to1zw{\yoko\hss#2}}%
536       \else\vtop{\hbox to1zw{\yoko#2\hss}}%
537       \fi\fi
538       \hskip\rensuji
539       \fi\else\hbox{#2}\fi
540 }
```

`\Rensuji` `\Rensuji` コマンドと `\prensuji` コマンドは、`\rensuji` コマンドで代用できます。

```
\prensuji 541 \let\Rensuji\rensuji
```

```
542 \let\prensuji\rensuji
```

漢数字

`\Kanji` `\Kanji` コマンドを定義します。`\Kanji` コマンドは `\Alph` と同じように、カウンタ `\@Kanji` に対してのみ使用することができます。

`\kanji` `\kanji` コマンドは、後続の半角数字を漢数字にします。`\kanji` 1989 のように指定をします。ただし、横組モードのときには、何もしません。つねに漢数字にしたい場合は、`\kansuji` プリミティブを使ってください。

後続の数字まで漢数字になってしまうバグを修正しました (Issue #33)。


```

543 \def\Kanji#1{\expandafter\@Kanji\csname c@#1\endcsname}
544 \def\@Kanji#1{\kansuji #1}
545 \def\kanji{\iftdir\expandafter\kansuji\fi}

```

傍点

`\boutenchar` `\bou` は、傍点を付けるコマンドです。

`\bou` 傍点として出力する文字は `\boutenchar` に指定します。この文字は、いつでも、横組用フォントが使われます。デフォルトは、EUC コード A1A2 (、) です。

```

546 \def\boutenchar{\char\euc"A1A2}

547 \DeclareRobustCommand\bou[1]{\ifvmode\leavevmode\fi\@bou#1\end}
548 \def\@bou#1{%
549   \ifx#1\end \let\next=\relax
550   \else
551     \iftdir\if@rotsw
552       \hbox to\z@{\vbox to\z@{\boxmaxdepth\maxdimen
553         \vss\moveleft-0.2zw\hbox{\boutenchar}\nointerlineskip
554         \hbox{\char\euc"A1A1}}\hss}\nobreak#1\relax
555     \else
556       \hbox to\z@{\vbox to\z@{\boxmaxdepth\maxdimen
557         \vss\moveleft0.2zw\hbox{\yoko\boutenchar}\nointerlineskip
558         \hbox{\char\euc"A1A1}}\hss}\nobreak#1\relax
559     \fi\else
560       \hbox to\z@{\vbox to\z@{%
561         \vss\moveleft-0.2zw\hbox{\yoko\boutenchar}\nointerlineskip
562         \hbox{\char\euc"A1A1}}\hss}\nobreak#1\relax
563     \fi
564     \let\next=\@bou
565   \fi\next}

```

下線

`\kasen` 下線を引くコマンドです。横組モードのときは、引数を `\underline` に渡します。縦組モードでも、回転モードの `\parbox` など使われたときには、やはり引数を `\underline` に渡します。これ以外の場合は、引数の上に直線を引きます。

```

566 \DeclareRobustCommand\kasen[1]{%
567   \iftdir\underline{#1}%
568   \else\if@rotsw\underline{#1}\else
569     \setbox\z@\hbox{#1}\leavevmode\raise.7zw
570     \hbox to\z@{\vrule\@width\wd\z@ \@depth\z@ \@height.4\p\hss}%
571     \box\z@
572   \fi\fi}

```

17.6 参照番号

参照番号の類を連数字で出力するように再定義します。itemize 環境などのリスト型のラベルについては、jarticle などのパッケージで定義しています。詳細は、jclasses.dtx を参照してください。

`\@eqnnum` これらは `\equation` コマンドで作成された数式に付加される番号です。 `ltmath.dtx`
`\@thecounter` で定義されています。

```
573 \def\@eqnnum{\reset@font\rmfamily \normalcolor
574   \iftdir\raise.25zh\hbox{\yoko(\theequation)}%
575   \else (\theequation)\fi}}
576 \def\@thecounter#1{\noexpand\rensuji{\noexpand\arabic{#1}}}
```

`\@thmcounter` `\newtheorem` コマンドで作成した環境で参照されるラベルです。 `ltthm.dtx` で定義
されています。

```
577 \def\@thmcounter#1{\noexpand\rensuji{\noexpand\arabic{#1}}}%
578 \end{package}
```

File f

pl209.dtx

18 DOCSTRIP 用モジュール

DOCSTRIP で以下のモジュール名を指定することで、対象となる部分を取り出すことができます。

pl209	pl209.def	ファイルを生成
oldfonts	oldfont.sty	を生成
style	jarticle	jarticle.sty ファイルを生成
	jbook	jbook.sty ファイルを生成
	jreport	jreport.sty ファイルを生成
	tarticle	tarticle.sty ファイルを生成
	tbook	tbook.sty ファイルを生成
	treport	treport.sty ファイルを生成

19 2.09 互換マクロ

2.09 用のコマンド定義ファイルがロードされたとき、メッセージを出力します。また、 \LaTeX の 2.09 コマンドマクロ定義をロードします。

```
1 <(*pl209)
2 \typeout{Entering pLaTeX 2.09 compatibility mode.}
3 \input{latex209.def}
4 </pl209>
```

フォント選択コマンドのトレースのために ptrace パッケージをロードします。

```
5 <oldfonts>\RequirePackage{oldfont}
6 <pl209 | oldfonts>\RequirePackage{ptrace}
```

\Rensuji $\text{pLaTeX 2}_{\epsilon}$ では、 \Rensuji 、 \prensuji の動作を \rensuji コマンドがカバーして \prensuji います。

```
7 <(*pl209)
8 \let\Rensuji\rensuji
9 \let\prensuji\rensuji
10 </pl209>
```

\@footnotemark 脚注の印を出力するマクロを、組み方向に応じて、脚注の方向が変わるようにし \@makefnmark ます。

```
11 <(*pl209)
12 \def\@footnotemark{\leavevmode
13 \ifhmode\edef\x@sf{\the\spacefactor}\fi
14 \ifdir\@makefnmark
15 \else\hbox to\z@{\hskip-.25zw\raise2\cht\@makefnmark\hss}\fi
16 \ifhmode\spacefactor\x@sf\fi\relax}
17 \def\@makefnmark{\hbox{\ifdir $\m@th^{\@thefnmark}$
```

```

18 \else\hbox{\yoko$\m@th^{\@thefnmark}$}\fi}
19 \pl209

20 (*pl209)
21 \fontencoding{JY1}
22 \fontfamily{mc}
23 \fontsize{10}{15}
24 \pl209

25 (*pl209 | oldfonts)
26 \DeclareSymbolFont{mincho}{JY1}{mc}{m}{n}
27 \DeclareSymbolFont{gothic}{JY1}{gt}{m}{n}
28 \DeclareSymbolFontAlphabet\mathmc{mincho}
29 \DeclareSymbolFontAlphabet\mathgt{gothic}
30 \SetSymbolFont{mincho}{bold}{JY1}{gt}{m}{n}
31 \jfam\symmincho

```

\mc と \gt は、和文フォントを変更しますが、欧文フォントには影響しません。

```

32 \DeclareRobustCommand\mc{%
33   \kanjiencoding{\kanjiencodingdefault}%
34   \kanjifamily{\mcdefault}%
35   \kanjiserries{\kanjiserriesdefault}%
36   \kanjishape{\kanjishapedefault}%
37   \selectfont\mathgroup\symmincho}
38 \DeclareRobustCommand\gt{%
39   \kanjiencoding{\kanjiencodingdefault}%
40   \kanjifamily{\gtdefault}%
41   \kanjiserries{\kanjiserriesdefault}%
42   \kanjishape{\kanjishapedefault}%
43   \selectfont\mathgroup\symgothic}

```

\bf コマンドは、和文フォントをゴシックにし、欧文フォントをボールドにします。

```

44 \DeclareRobustCommand\bf{\normalfont\bfseries\mathgroup\symbol0\jfam\symgothic}

```

\rm, \sf, \sl, \sc, \it, \tt の各コマンドを、欧文ファミリだけをデフォルトフォントから属性を変更するようにし、和文フォントは影響を受けないように修正します。

```

45 \DeclareRobustCommand\roman@normal{%
46   \romanencoding{\encodingdefault}%
47   \romanfamily{\familydefault}%
48   \romanseries{\seriesdefault}%
49   \romanshape{\shapedefault}%
50   \selectfont\ignorespaces}
51 \DeclareRobustCommand\rm{\roman@normal\rmfamily\mathgroup\symoperators}
52 \DeclareRobustCommand\sffamily{\roman@normal\sffamily\mathgroup\symsans}
53 \DeclareRobustCommand\sl{\roman@normal\slshape\mathgroup\symslanted}
54 \DeclareRobustCommand\sc{\roman@normal\scshape\mathgroup\symsmallcaps}
55 \DeclareRobustCommand\it{\roman@normal\itshape\mathgroup\symitalic}
56 \DeclareRobustCommand\tt{\roman@normal\ttfamily\mathgroup\syntypewriter}

```

\em \em コマンドで、和文フォントも \gt に切り替えるようにしました。

```

57 \DeclareRobustCommand\em{%
58   \@nomath\em
59   \ifdim \fontdimen\@ne\font>\z@\mc\rm\else\gt\it\fi}
60 \pl209 | oldfonts)

```

```

61 < *pl209>
62 \let\mcfam\symmincho
63 \let\gtfam\symgothic
64 \renewcommand\vpt {\edef\fs{\@vpt}\rm\mc}
65 \renewcommand\vipt {\edef\fs{\@vipt}\rm\mc}
66 \renewcommand\vipt {\edef\fs{\@vipt}\rm\mc}
67 \renewcommand\viipt{\edef\fs{\@viipt}\rm\mc}
68 \renewcommand\ixpt {\edef\fs{\@ixpt}\rm\mc}
69 \renewcommand\xpt {\edef\fs{\@xpt}\rm\mc}
70 \renewcommand\xipt {\edef\fs{\@xipt}\rm\mc}
71 \renewcommand\xiipt {\edef\fs{\@xiipt}\rm\mc}
72 \renewcommand\xivpt {\edef\fs{\@xivpt}\rm\mc}
73 \renewcommand\xvipt{\edef\fs{\@xvipt}\rm\mc}
74 \renewcommand\xxpt {\edef\fs{\@xxpt}\rm\mc}
75 \renewcommand\xxvpt {\edef\fs{\@xxvpt}\rm\mc}
76 < /pl209>

```

そして、最後に pl209.cfg というファイルがあれば、それをロードします。

```

77 < pl209>\InputIfFileExists{pl209.cfg}{ }{}

```

20 スタイルファイル

以下は、p_{La}T_EX 2.09 での標準スタイルファイルです。p_{La}T_EX 2_ε のクラスファイルをロードするようにしています。

```

78 < *style>
79 < *article | jbook | jreport | tarticle | tbook | treport>
80 \NeedsTeXFormat{pLaTeX2e}
81 < /article | jbook | jreport | tarticle | tbook | treport>
82 < *article>
83 \@obsoletedefile{jarticle.cls}{jarticle.sty}
84 \LoadClass{jarticle}
85 < /article>
86 < *tarticle>
87 \@obsoletedefile{tarticle.cls}{tarticle.sty}
88 \LoadClass{tarticle}
89 < /tarticle>
90 < *jbook>
91 \@obsoletedefile{jbook.cls}{jbook.sty}
92 \LoadClass{jbook}
93 < /jbook>
94 < *tbook>
95 \@obsoletedefile{tbook.cls}{tbook.sty}
96 \LoadClass{tbook}
97 < /tbook>
98 < *jreport>
99 \@obsoletedefile{jreport.cls}{jreport.sty}
100 \LoadClass{jreport}
101 < /jreport>
102 < *treport>
103 \@obsoletedefile{treport.cls}{treport.sty}
104 \LoadClass{treport}
105 < /treport>

```

106 $\langle \text{style} \rangle$

File g

kinsoku.dtx

このファイルは、禁則と文字間スペースの設定について説明をしています。日本語 TeX の機能についての詳細は、『日本語 TeX テクニカルブック I』を参照してください。

なお、このファイルのコード部分は、以前のバージョンで配布された kinsoku.tex と同一です。

```
1 (*!core)
```

21 禁則

ある文字を行頭禁則の対象にするには、`\prebreakpenalty` に正の値を指定します。ある文字を行末禁則の対象にするには、`\postbreakpenalty` に正の値を指定します。数値が大きいほど、行頭、あるいは行末で改行されにくくなります。

21.1 半角文字に対する禁則

ここでは、半角文字に対する禁則の設定を行なっています。

```
2 \prebreakpenalty'!=10000
3 \prebreakpenalty'"=10000
4 \postbreakpenalty'\#=500
5 \postbreakpenalty'\$=500
6 \prebreakpenalty'\%=500
7 \prebreakpenalty'\&=500
8 \postbreakpenalty'\ '=10000
9 \prebreakpenalty'\ ' =10000
10 \prebreakpenalty'\ ' =10000
11 \postbreakpenalty'\ ' =10000
12 \prebreakpenalty'\*=500
13 \prebreakpenalty'\+=500
14 \prebreakpenalty'\-=10000
15 \prebreakpenalty'\.=10000
16 \prebreakpenalty'\,=10000
17 \prebreakpenalty'\ /=500
18 \prebreakpenalty'\;=10000
19 \prebreakpenalty'\?=10000
20 \prebreakpenalty'\:=10000
21 \prebreakpenalty'\]=10000
22 \postbreakpenalty'\ [=10000
```

21.2 全角文字に対する禁則

ここでは、全角文字に対する禁則の設定を行なっています。

```
23 \prebreakpenalty', =10000
24 \prebreakpenalty'。 =10000
25 \prebreakpenalty', =10000
```

```

26 \prebreakpenalty' . =10000
27 \prebreakpenalty' · =10000
28 \prebreakpenalty' : =10000
29 \prebreakpenalty' ; =10000
30 \prebreakpenalty' ? =10000
31 \prebreakpenalty' ! =10000
32 \prebreakpenalty\jis"212B=10000
33 \prebreakpenalty\jis"212C=10000
34 \prebreakpenalty\jis"212D=10000
35 \postbreakpenalty\jis"212E=10000
36 \prebreakpenalty\jis"2139=10000
37 \prebreakpenalty\jis"2144=250
38 \prebreakpenalty\jis"2145=250
39 \postbreakpenalty\jis"2146=10000
40 \prebreakpenalty\jis"2147=5000
41 \postbreakpenalty\jis"2148=5000
42 \prebreakpenalty\jis"2149=5000
43 \prebreakpenalty' ) =10000
44 \postbreakpenalty' ( =10000
45 \prebreakpenalty' } =10000
46 \postbreakpenalty' { =10000
47 \prebreakpenalty' ] =10000
48 \postbreakpenalty' [ =10000
49 \postbreakpenalty' ' =10000
50 \prebreakpenalty' ' =10000
51 \postbreakpenalty\jis"214C=10000
52 \prebreakpenalty\jis"214D=10000
53 \postbreakpenalty\jis"2152=10000
54 \prebreakpenalty\jis"2153=10000
55 \postbreakpenalty\jis"2154=10000
56 \prebreakpenalty\jis"2155=10000
57 \postbreakpenalty\jis"2156=10000
58 \prebreakpenalty\jis"2157=10000
59 \postbreakpenalty\jis"2158=10000
60 \prebreakpenalty\jis"2159=10000
61 \postbreakpenalty\jis"215A=10000
62 \prebreakpenalty\jis"215B=10000
63 \prebreakpenalty' — =10000
64 \prebreakpenalty' + =200
65 \prebreakpenalty' - =200
66 \prebreakpenalty' = =200
67 \postbreakpenalty' # =200
68 \postbreakpenalty' $ =200
69 \prebreakpenalty' % =200
70 \prebreakpenalty' & =200
71 \prebreakpenalty' あ =150
72 \prebreakpenalty' い =150
73 \prebreakpenalty' う =150
74 \prebreakpenalty' え =150
75 \prebreakpenalty' お =150
76 \prebreakpenalty' つ =150
77 \prebreakpenalty' や =150
78 \prebreakpenalty' む =150
79 \prebreakpenalty' よ =150

```



```

80 \prebreakpenalty\jis"246E=150
81 \prebreakpenalty'ア=150
82 \prebreakpenalty'イ=150
83 \prebreakpenalty'ウ=150
84 \prebreakpenalty'エ=150
85 \prebreakpenalty'オ=150
86 \prebreakpenalty'ツ=150
87 \prebreakpenalty'ヤ=150
88 \prebreakpenalty'ユ=150
89 \prebreakpenalty'ヨ=150
90 \prebreakpenalty\jis"256E=150
91 \prebreakpenalty\jis"2575=150
92 \prebreakpenalty\jis"2576=150

```

22 文字間のスペース

ある英字の前後と、その文字に隣合う漢字に挿入されるスペースを制御するには、`\xspcode` を用います。

ある漢字の前後と、その文字に隣合う英字に挿入されるスペースを制御するには、`\inhibitxspcode` を用います。

22.1 ある英字と前後の漢字の間の制御

ここでは、英字に対する設定を行なっています。

指定する数値とその意味は次のとおりです。

- 0 前後の漢字の間での処理を禁止する。
- 1 直前の漢字との間にのみ、スペースの挿入を許可する。
- 2 直後の漢字との間にのみ、スペースの挿入を許可する。
- 3 前後の漢字との間でのスペースの挿入を許可する。

```

93 \xspcode' (=1
94 \xspcode')=2
95 \xspcode'[ =1
96 \xspcode']=2
97 \xspcode' '=1
98 \xspcode'' =2
99 \xspcode': =2
100 \xspcode'; =2
101 \xspcode', =2
102 \xspcode'. =2

```

T1 などの 8 ビットフォントエンコーディングで 128-255 の文字は欧文文字ですので、周囲の和文文字との間に `\xkanjiskip` が挿入される必要があります。そこで、奥村さんの `jsclasses` や田中さんの `upLATEX` と同等の対処をします。

```

103 \xspcode"80=3
104 \xspcode"81=3
105 \xspcode"82=3
106 \xspcode"83=3

```

107 \xspcode"84=3
108 \xspcode"85=3
109 \xspcode"86=3
110 \xspcode"87=3
111 \xspcode"88=3
112 \xspcode"89=3
113 \xspcode"8A=3
114 \xspcode"8B=3
115 \xspcode"8C=3
116 \xspcode"8D=3
117 \xspcode"8E=3
118 \xspcode"8F=3
119 \xspcode"90=3
120 \xspcode"91=3
121 \xspcode"92=3
122 \xspcode"93=3
123 \xspcode"94=3
124 \xspcode"95=3
125 \xspcode"96=3
126 \xspcode"97=3
127 \xspcode"98=3
128 \xspcode"99=3
129 \xspcode"9A=3
130 \xspcode"9B=3
131 \xspcode"9C=3
132 \xspcode"9D=3
133 \xspcode"9E=3
134 \xspcode"9F=3
135 \xspcode"A0=3
136 \xspcode"A1=3
137 \xspcode"A2=3
138 \xspcode"A3=3
139 \xspcode"A4=3
140 \xspcode"A5=3
141 \xspcode"A6=3
142 \xspcode"A7=3
143 \xspcode"A8=3
144 \xspcode"A9=3
145 \xspcode"AA=3
146 \xspcode"AB=3
147 \xspcode"AC=3
148 \xspcode"AD=3
149 \xspcode"AE=3
150 \xspcode"AF=3
151 \xspcode"B0=3
152 \xspcode"B1=3
153 \xspcode"B2=3
154 \xspcode"B3=3
155 \xspcode"B4=3
156 \xspcode"B5=3
157 \xspcode"B6=3
158 \xspcode"B7=3
159 \xspcode"B8=3
160 \xspcode"B9=3

161 \xspcode"BA=3
162 \xspcode"BB=3
163 \xspcode"BC=3
164 \xspcode"BD=3
165 \xspcode"BE=3
166 \xspcode"BF=3
167 \xspcode"C0=3
168 \xspcode"C1=3
169 \xspcode"C2=3
170 \xspcode"C3=3
171 \xspcode"C4=3
172 \xspcode"C5=3
173 \xspcode"C6=3
174 \xspcode"C7=3
175 \xspcode"C8=3
176 \xspcode"C9=3
177 \xspcode"CA=3
178 \xspcode"CB=3
179 \xspcode"CC=3
180 \xspcode"CD=3
181 \xspcode"CE=3
182 \xspcode"CF=3
183 \xspcode"D0=3
184 \xspcode"D1=3
185 \xspcode"D2=3
186 \xspcode"D3=3
187 \xspcode"D4=3
188 \xspcode"D5=3
189 \xspcode"D6=3
190 \xspcode"D7=3
191 \xspcode"D8=3
192 \xspcode"D9=3
193 \xspcode"DA=3
194 \xspcode"DB=3
195 \xspcode"DC=3
196 \xspcode"DD=3
197 \xspcode"DE=3
198 \xspcode"DF=3
199 \xspcode"E0=3
200 \xspcode"E1=3
201 \xspcode"E2=3
202 \xspcode"E3=3
203 \xspcode"E4=3
204 \xspcode"E5=3
205 \xspcode"E6=3
206 \xspcode"E7=3
207 \xspcode"E8=3
208 \xspcode"E9=3
209 \xspcode"EA=3
210 \xspcode"EB=3
211 \xspcode"EC=3
212 \xspcode"ED=3
213 \xspcode"EE=3
214 \xspcode"EF=3

```

215 \xspcode"F0=3
216 \xspcode"F1=3
217 \xspcode"F2=3
218 \xspcode"F3=3
219 \xspcode"F4=3
220 \xspcode"F5=3
221 \xspcode"F6=3
222 \xspcode"F7=3
223 \xspcode"F8=3
224 \xspcode"F9=3
225 \xspcode"FA=3
226 \xspcode"FB=3
227 \xspcode"FC=3
228 \xspcode"FD=3
229 \xspcode"FE=3
230 \xspcode"FF=3

```

22.2 ある漢字と前後の英字の間の制御

ここでは、漢字に対する設定を行なっています。

指定する数値とその意味は次のとおりです。

- 0 前後の英字との間にスペースを挿入することを禁止する。
- 1 直前の英字との間にスペースを挿入することを禁止する。
- 2 直後の英字との間にスペースを挿入することを禁止する。
- 3 前後の英字との間でのスペースの挿入を許可する。

```

231 \inhibitxspcode‘, =1
232 \inhibitxspcode‘。 =1
233 \inhibitxspcode‘, =1
234 \inhibitxspcode‘. =1
235 \inhibitxspcode‘: =1
236 \inhibitxspcode‘; =1
237 \inhibitxspcode‘? =1
238 \inhibitxspcode‘! =1
239 \inhibitxspcode‘) =1
240 \inhibitxspcode‘( =2
241 \inhibitxspcode‘] =1
242 \inhibitxspcode‘[ =2
243 \inhibitxspcode‘} =1
244 \inhibitxspcode‘{ =2
245 \inhibitxspcode‘‘ =2
246 \inhibitxspcode‘’ =1
247 \inhibitxspcode‘“ =2
248 \inhibitxspcode‘” =1
249 \inhibitxspcode‘[ =2
250 \inhibitxspcode‘] =1
251 \inhibitxspcode‘< =2
252 \inhibitxspcode‘> =1
253 \inhibitxspcode‘《 =2
254 \inhibitxspcode‘》 =1
255 \inhibitxspcode‘「 =2

```

```

256 \inhibitxspcode'」=1
257 \inhibitxspcode'『=2
258 \inhibitxspcode'』=1
259 \inhibitxspcode'【=2
260 \inhibitxspcode'】=1
261 \inhibitxspcode'—=0
262 \inhibitxspcode'～=0
263 \inhibitxspcode'…=0
264 \inhibitxspcode'¥=0
265 \inhibitxspcode'°=1
266 \inhibitxspcode' '=1
267 \inhibitxspcode'"=1
268 </plcore>

```

File h jclasses.dtx

このファイルは、pL^AT_EX 2_ε の標準クラスファイルです。DOCSTRIP プログラムによって、横組用のクラスファイルと縦組用のクラスファイルを作成することができます。

次に DOCSTRIP プログラムのためのオプションを示します。

オプション	意味
article	article クラスを生成
report	report クラスを生成
book	book クラスを生成
10pt	10pt サイズの設定を生成
11pt	11pt サイズの設定を生成
12pt	12pt サイズの設定を生成
bk	book クラス用のサイズの設定を生成
tate	縦組用の設定を生成
yoko	横組用の設定を生成

23 オプションスイッチ

ここでは、後ほど使用するいくつかのコマンドやスイッチを定義しています。

`\c@paper` 用紙サイズを示すために使います。A4, A5, B4, B5 用紙はそれぞれ、1, 2, 3, 4 として表されます。

```
1 <{*article | report | book}>
2 \newcounter{c@paper}
```

`\if@landscape` 用紙を横向きにするかどうかのスイッチです。デフォルトは、縦向きです。

```
3 \newif\if@landscape \@landscapefalse
```

`\@ptsize` 組版をするポイント数の一の位を保存するために使います。0, 1, 2 のいずれかです。

```
4 \newcommand{\@ptsize}{}
```

`\if@restonecol` 二段組時に用いるテンポラリスイッチです。

```
5 \newif\if@restonecol
```

`\if@titlepage` タイトルページやアブストラクト（概要）を独立したページにするかどうかのスイッチです。report と book スタイルのデフォルトでは、独立したページになります。

```
6 \newif\if@titlepage
7 <article>\@titlepagefalse
8 <report | book>\@titlepagetrue
```

`\if@openright` chapter レベルを右ページからはじめるかどうかのスイッチです。横組では奇数ページ、縦組では偶数ページから始まることになります。report クラスのデフォルトは、“no” です。book クラスのデフォルトは、“yes” です。

```
9 \!article\newif\if@openright
```

`\if@openleft` chapter レベルを左ページからはじめるかどうかのスイッチです。日本語 $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ 開発コミュニティ版で新たに追加されました。横組では偶数ページ、縦組では奇数ページから始まることになります。report クラスと book クラスの両方で、デフォルトは “no” です。

```
10 \!article\newif\if@openleft
```

`\if@mainmatter` スイッチ `\@mainmatter` が真の場合、本文を処理しています。このスイッチが偽の場合は、`\chapter` コマンドは見出し番号を出力しません。

```
11 \!book\newif\if@mainmatter \@mainmattertrue
```

```
\hour
\minute 12 \hour\time \divide\hour by 60\relax
13 \@tempcnta\hour \multiply\@tempcnta 60\relax
14 \minute\time \advance\minute-\@tempcnta
```

`\if@stysize` $\mathrm{pL}_{\mathrm{T}}\mathrm{X}_{2_{\epsilon}}$ 2.09 互換モードで、スタイルオプションに `a4j,a5p` などが指定されたときの動作をエミュレートするためのフラグです。

```
15 \newif\if@stysize \@stysizefalse
```

`\if@enablejfam` 日本語ファミリを宣言するために用いるフラグです。

```
16 \newif\if@enablejfam \@enablejfamtrue
```

和欧文両対応の数式文字コマンドを有効にするときに用いるフラグです。マクロの展開順序が複雑になるのを避けるため、デフォルトでは `false` としてあります。

```
17 \newif\if@mathrmmc \@mathrmmcfalse
```

24 オプションの宣言

ここでは、クラスオプションの宣言を行なっています。

24.1 用紙オプション

用紙サイズを指定するオプションです。

```
18 \DeclareOption{a4paper}{\setcounter{@paper}{1}%
19 \setlength\paperheight {297mm}%
20 \setlength\paperwidth {210mm}}
21 \DeclareOption{a5paper}{\setcounter{@paper}{2}%
22 \setlength\paperheight {210mm}
23 \setlength\paperwidth {148mm}}
24 \DeclareOption{b4paper}{\setcounter{@paper}{3}%
25 \setlength\paperheight {364mm}}
```

```

26 \setlength\paperwidth {257mm}
27 \DeclareOption{b5paper}{\setcounter{@paper}{4}%
28 \setlength\paperheight {257mm}
29 \setlength\paperwidth {182mm}}

```

ドキュメントクラスに、以下のオプションを指定すると、通常よりもテキストを組み立てる領域の広いスタイルとすることができます。

```

30 %
31 \DeclareOption{a4j}{\setcounter{@paper}{1}\@stysizetrue
32 \setlength\paperheight {297mm}%
33 \setlength\paperwidth {210mm}}
34 \DeclareOption{a5j}{\setcounter{@paper}{2}\@stysizetrue
35 \setlength\paperheight {210mm}
36 \setlength\paperwidth {148mm}}
37 \DeclareOption{b4j}{\setcounter{@paper}{3}\@stysizetrue
38 \setlength\paperheight {364mm}
39 \setlength\paperwidth {257mm}}
40 \DeclareOption{b5j}{\setcounter{@paper}{4}\@stysizetrue
41 \setlength\paperheight {257mm}
42 \setlength\paperwidth {182mm}}
43 %
44 \DeclareOption{a4p}{\setcounter{@paper}{1}\@stysizetrue
45 \setlength\paperheight {297mm}%
46 \setlength\paperwidth {210mm}}
47 \DeclareOption{a5p}{\setcounter{@paper}{2}\@stysizetrue
48 \setlength\paperheight {210mm}
49 \setlength\paperwidth {148mm}}
50 \DeclareOption{b4p}{\setcounter{@paper}{3}\@stysizetrue
51 \setlength\paperheight {364mm}
52 \setlength\paperwidth {257mm}}
53 \DeclareOption{b5p}{\setcounter{@paper}{4}\@stysizetrue
54 \setlength\paperheight {257mm}
55 \setlength\paperwidth {182mm}}

```

24.2 サイズオプション

基準となるフォントの大きさを指定するオプションです。

```

56 \if@compatibility
57 \renewcommand{\@ptsize}{0}
58 \else
59 \DeclareOption{10pt}{\renewcommand{\@ptsize}{0}}
60 \fi
61 \DeclareOption{11pt}{\renewcommand{\@ptsize}{1}}
62 \DeclareOption{12pt}{\renewcommand{\@ptsize}{2}}

```

24.3 横置きオプション

このオプションが指定されると、用紙の縦と横の長さを入れ換えます。

```

63 \DeclareOption{landscape}{\@landscapetrue
64 \setlength\@tempdima{\paperheight}%
65 \setlength\paperheight{\paperwidth}%
66 \setlength\paperwidth{\@tempdima}}

```


24.4 トンボオプション

tombow オプションが指定されると、用紙サイズに合わせてトンボを出力します。このとき、トンボの脇に DVI を作成した日付が出力されます。作成日付の出力を抑制するには、tombow ではなく、tombo と指定をします。

ジョブ情報の書式は元々 filename : 2017/3/5(13:3) のような書式でしたが、jsclasses にあわせて桁数固定の filename (2017-03-05 13:03) に直しました。

```
67 \DeclareOption{tombow}{%
68   \tombowtrue \tombowdatetrue
69   \setlength{\@tombowwidth}{.1\p@}%
70   \@bannertoken{%
71     \jobname\space(\number\year-\two@digits\month-\two@digits\day
72     \space\two@digits\hour:\two@digits\minute)}%
73   \maketombowbox}
74 \DeclareOption{tombo}{%
75   \tombowtrue \tombowdatefalse
76   \setlength{\@tombowwidth}{.1\p@}%
77   \maketombowbox}
```

24.5 面付けオプション

このオプションが指定されると、トンボオプションを指定したときと同じ位置に文章を出力します。作成した DVI をフィルムに面付け出力する場合などに指定をします。

```
78 \DeclareOption{mentuke}{%
79   \tombowtrue \tombowdatefalse
80   \setlength{\@tombowwidth}{\z@}%
81   \maketombowbox}
```

24.6 組方向オプション

このオプションが指定されると、縦組で組版をします。

```
82 \DeclareOption{tate}{%
83   \AtBeginDocument{\tate\message{《縦組モード》}%
84     \adjustbaseline}%
85 }
```

24.7 両面、片面オプション

twoside オプションが指定されると、両面印字出力に適した整形を行ないます。

```
86 \DeclareOption{oneside}{\@twosidefalse}
87 \DeclareOption{twoside}{\@twosidetrue}
```

24.8 二段組オプション

二段組にするかどうかのオプションです。

```
88 \DeclareOption{onecolumn}{\@twocolumnfalse}
89 \DeclareOption{twocolumn}{\@twocolumntrue}
```

24.9 表題ページオプション

@titlepage が真の場合、表題を独立したページに出力します。

```
90 \DeclareOption{titlepage}{\@titlepagetrue}
91 \DeclareOption{notitlepage}{\@titlepagefalse}
```

24.10 右左起こしオプション

chapter を右ページあるいは左ページからはじめるかどうかを指定するオプションです。openleft オプションは日本語 T_EX 開発コミュニティによって追加されました。

```
92 \!article\if@compatibility
93 \book\@openrighttrue
94 \!article\else
95 \!article\DeclareOption{openright}{\@openrighttrue\@openleftfalse}
96 \!article\DeclareOption{openleft}{\@openlefttrue\@openrightfalse}
97 \!article\DeclareOption{openany}{\@openrightfalse\@openleftfalse}
98 \!article\fi
```

24.11 数式のオプション

leqno を指定すると、数式番号を数式の左側に出力します。fleqn を指定するとディスプレイ数式を左揃えで出力します。

```
99 \DeclareOption{leqno}{\input{leqno.clo}}
100 \DeclareOption{fleqn}{\input{fleqn.clo}}
```

24.12 参考文献のオプション

参考文献一覧を“オープンスタイル”の書式で出力します。これは各ブロックが改行で区切られ、\bibindent のインデントが付く書式です。

```
101 \DeclareOption{openbib}{%
```

参考文献環境内の最初のいくつかのフックを満たします。

```
102 \AtEndOfPackage{%
103 \renewcommand\@openbib@code{%
104 \advance\leftmargin\bibindent
105 \itemindent -\bibindent
106 \listparindent \itemindent
107 \parsep \z@
108 }%
```

そして、\newblock を再定義します。

```
109 \renewcommand\newblock{\par}}
```

24.13 日本語ファミリ宣言の抑制、和欧文両対応の数式文字

p_AT_EX 2_ε は、このあと、数式モードで直接、日本語を記述できるように数式ファミリを宣言します。しかし、T_EX で扱える数式ファミリの数が 16 個なので、その他のパッケージと組み合わせた場合、数式ファミリを宣言する領域を超えてしまう

場合があるかもしれません。そのときには、残念ですが、そのパッケージか、数式内に直接、日本語を記述するのか、どちらかを断念しなければなりません。このクラスオプションは、数式内に日本語を記述するのをあきらめる場合に用います。

`disablejfam` オプションを指定しても `\textmc` や `\textgt` などを用いて、数式内に日本語を記述することは可能です。

日本語 $T_{\text{E}}X$ 開発コミュニティによる補足：コミュニティ版 $\text{p}T_{\text{E}}X$ の 2016/11/29 以降の版では、 $\text{e-p}T_{\text{E}}X$ の拡張機能（通称「旧 FAM256 パッチ」）が利用可能な場合に、 $\text{I}T_{\text{E}}X$ の機能で宣言できる数式ファミリー（数式アルファベット）の上限を 256 個に増やしています。したがって、新しい環境では `disablejfam` を指定しなくても上限を超えることが起きにくくなっています。

`mathrmmc` オプションは、`\mathrm` と `\mathbf` を和欧文両対応にするためのクラスオプションです。

```
110 \if@compatibility
111   \@mathrmctrue
112 \else
113   \DeclareOption{disablejfam}{\@enablejfamfalse}
114   \DeclareOption{mathrmmc}{\@mathrmctrue}
115 \fi
```

24.14 ドラフトオプション

`draft` オプションを指定すると、オーバフルボックスの起きた箇所に、5pt の罫線が引かれます。

```
116 \DeclareOption{draft}{\setlength\overfullrule{5pt}}
117 \DeclareOption{final}{\setlength\overfullrule{0pt}}
118 \langle article | report | book \rangle
```

24.15 オプションの実行

オプションの実行、およびサイズクラスのロードを行ないます。

```
119 \langle *article | report | book \rangle
120 \langle *article \rangle
121 \langle tate \rangle \ExecuteOptions{a4paper,10pt,oneside,onecolumn,final,tate}
122 \langle yoko \rangle \ExecuteOptions{a4paper,10pt,oneside,onecolumn,final}
123 \langle /article \rangle
124 \langle *report \rangle
125 \langle tate \rangle \ExecuteOptions{a4paper,10pt,oneside,onecolumn,final,openany,tate}
126 \langle yoko \rangle \ExecuteOptions{a4paper,10pt,oneside,onecolumn,final,openany}
127 \langle /report \rangle
128 \langle *book \rangle
129 \langle tate \rangle \ExecuteOptions{a4paper,10pt,twoside,onecolumn,final,openright,tate}
130 \langle yoko \rangle \ExecuteOptions{a4paper,10pt,twoside,onecolumn,final,openright}
131 \langle /book \rangle
132 \ProcessOptions\relax
133 \langle book & tate \rangle \input{tbk1\@ptsize.clo}
134 \langle !book & tate \rangle \input{tsize1\@ptsize.clo}
135 \langle book & yoko \rangle \input{jbk1\@ptsize.clo}
136 \langle !book & yoko \rangle \input{jsize1\@ptsize.clo}
```

縦組用クラスファイルの場合は、ここで `plext.sty` も読み込みます。

```
137 <tate>\RequirePackage{plext}
138 </article | report | book>
```

25 フォント

ここでは、 \LaTeX のフォントサイズコマンドの定義をしています。フォントサイズコマンドの定義は、次のコマンドを用います。

```
\setfontsize\size<font-size>\baselineskip
```

$\langle font-size \rangle$ これから使用する、フォントの実際の大きさです。

$\langle baselineskip \rangle$ 選択されるフォントサイズ用の通常の $\backslash baselineskip$ の値です（実際は、 $\backslash baselinestretch * \langle baselineskip \rangle$ の値です）。

数値コマンドは、次のように \LaTeX カーネルで定義されています。

```
\@vpt      5      \@vipt    6      \@viipt    7
\@viiipt   8      \@ixpt    9      \@xpt      10
\@xipt     10.95  \@xiipt   12      \@xivpt    14.4
...
```

$\backslash normalsize$ 基本サイズとするユーザレベルのコマンドは $\backslash normalsize$ です。 \LaTeX の内部では $\backslash @normalsize$ $\backslash @normalsize$ を使用します。

$\backslash normalsize$ マクロは、 $\backslash abovedisplayskip$ と $\backslash abovedisplayshortskip$ 、および $\backslash belowdisplayshortskip$ の値も設定をします。 $\backslash belowdisplayskip$ は、つねに $\backslash abovedisplayskip$ と同値です。

また、リスト環境のトップレベルのパラメータは、つねに $\backslash @listI$ で与えられます。

```
139 <*10pt | 11pt | 12pt>
140 \renewcommand{\normalsize}{%
141   <10pt & yoko>    \@setfontsize\normalsize\@xpt{15}%
142   <11pt & yoko>    \@setfontsize\normalsize\@xipt{15.5}%
143   <12pt & yoko>    \@setfontsize\normalsize\@xiipt{16.5}%
144   <10pt & tate>    \@setfontsize\normalsize\@xpt{17}%
145   <11pt & tate>    \@setfontsize\normalsize\@xipt{17}%
146   <12pt & tate>    \@setfontsize\normalsize\@xiipt{18}%
147 <*10pt>
148   \abovedisplayskip 10\p@ \@plus2\p@ \@minus5\p@
149   \abovedisplayshortskip \z@ \@plus3\p@
150   \belowdisplayshortskip 6\p@ \@plus3\p@ \@minus3\p@
151 </10pt>
152 <*11pt>
153   \abovedisplayskip 11\p@ \@plus3\p@ \@minus6\p@
154   \abovedisplayshortskip \z@ \@plus3\p@
155   \belowdisplayshortskip 6.5\p@ \@plus3.5\p@ \@minus3\p@
156 </11pt>
```

```

157 (*12pt)
158 \abovedisplayskip 12\p@ \@plus3\p@ \@minus7\p@
159 \abovedisplayshortskip \z@ \@plus3\p@
160 \belowdisplayshortskip 6.5\p@ \@plus3.5\p@ \@minus3\p@
161 (/12pt)
162 \belowdisplayskip \abovedisplayskip
163 \let\@listi\@listI}

```

ここで、ノーマルフォントを選択し、初期化をします。このとき、縦組モードならば、デフォルトのエンコードを変更します。

```

164 (tate)\def\kanjiencodingdefault{JT1}%
165 (tate)\kanjiencoding{\kanjiencodingdefault}%
166 \normalsize

```

\normalsize を robust にします。すぐ上で \DeclareRobustCommand とせずに、カーネルの定義を \renewcommand した後に \MakeRobust を使っている理由は、ログに LaTeX Info: Redefining \normalsize on input line ... というメッセージを出したくないからです。ただし、latexrelease パッケージで 2015/01/01 より昔の日付に巻き戻っている場合は \MakeRobust が定義されていません。

```

167 \ifx\MakeRobust\@undefined \else
168   \MakeRobust\normalsize
169 \fi

```

\Cht 基準となる長さの設定をします。これらのパラメータは plfonts.dtx で定義されて \Cdp います。基準とする文字を「全角空白」(EUC コード 0xA1A1) から「漢」(JIS コード 0x3441) へ変更しました。

```

\Cvs 170 \setbox0\hbox{\char\jis"3441}%
\Cht 171 \setlength\Cht{\ht0}
\Cdp 172 \setlength\Cdp{\dp0}
\Cwd 173 \setlength\Cwd{\wd0}
174 \setlength\Cvs{\baselineskip}
175 \setlength\Chs{\wd0}
176 \setbox0=\box\voidb@x

```

\small \small コマンドの定義は、\normalsize に似ています。こちらはカーネルで未定義なので、直接 \DeclareRobustCommand で定義します。

```

177 \DeclareRobustCommand{\small}{%
178 (*10pt)
179 \@setfontsize\small\@ixpt{11}%
180 \abovedisplayskip 8.5\p@ \@plus3\p@ \@minus4\p@
181 \abovedisplayshortskip \z@ \@plus2\p@
182 \belowdisplayshortskip 4\p@ \@plus2\p@ \@minus2\p@
183 \def\@listi{\leftmargin\leftmarginI
184             \topsep 4\p@ \@plus2\p@ \@minus2\p@
185             \parsep 2\p@ \@plus\p@ \@minus\p@
186             \itemsep \parsep}%
187 (/10pt)
188 (*11pt)
189 \@setfontsize\small\@xpt\@xipt
190 \abovedisplayskip 10\p@ \@plus2\p@ \@minus5\p@
191 \abovedisplayshortskip \z@ \@plus3\p@

```

```

192 \belowdisplayshortskip 6\p@ \@plus3\p@ \@minus3\p@
193 \def\@listif\leftmargin\leftmargini
194     \topsep 6\p@ \@plus2\p@ \@minus2\p@
195     \parsep 3\p@ \@plus2\p@ \@minus\p@
196     \itemsep \parsep}%
197 \
```

`\footnotesize` `\footnotesize` コマンドの定義は、`\normalsize` に似ています。こちらも直接 `\DeclareRobustCommand` で定義します。

```

209 \DeclareRobustCommand\footnotesize{%
210 (*10pt)
211 \setfontsize\footnotesize\@viipt{9.5}%
212 \abovedisplayskip 6\p@ \@plus2\p@ \@minus4\p@
213 \abovedisplayshortskip \z@ \@plus\p@
214 \belowdisplayshortskip 3\p@ \@plus\p@ \@minus2\p@
215 \def\@listif\leftmargin\leftmargini
216     \topsep 3\p@ \@plus\p@ \@minus\p@
217     \parsep 2\p@ \@plus\p@ \@minus\p@
218     \itemsep \parsep}%
219 \
```

`\scriptsize` これらは先ほどのマクロよりも簡単です。これらはフォントサイズを変更するだけ

`\tiny`

`\large` File h: jclasses.dtx

168

`\Large`

`\LARGE`

`\huge`

`\Huge`

で、リスト環境とディスプレイ数式のパラメータは変更しません。

```
241 (*10pt)
242 \DeclareRobustCommand{\scriptsize}{\@setfontsize\scriptsize\@viipt\@viipt}
243 \DeclareRobustCommand{\tiny}{\@setfontsize\tiny\@vpt\@vpt}
244 \DeclareRobustCommand{\large}{\@setfontsize\large\@xiipt{17}}
245 \DeclareRobustCommand{\Large}{\@setfontsize\Large\@xivpt{21}}
246 \DeclareRobustCommand{\LARGE}{\@setfontsize\LARGE\@xxviipt{25}}
247 \DeclareRobustCommand{\huge}{\@setfontsize\huge\@xxpt{28}}
248 \DeclareRobustCommand{\Huge}{\@setfontsize\Huge\@xxvpt{33}}
249 (/10pt)
250 (*11pt)
251 \DeclareRobustCommand{\scriptsize}{\@setfontsize\scriptsize\@viiipt{9.5}}
252 \DeclareRobustCommand{\tiny}{\@setfontsize\tiny\@vipt\@vipt}
253 \DeclareRobustCommand{\large}{\@setfontsize\large\@xipt{17}}
254 \DeclareRobustCommand{\Large}{\@setfontsize\Large\@xivpt{21}}
255 \DeclareRobustCommand{\LARGE}{\@setfontsize\LARGE\@xxviipt{25}}
256 \DeclareRobustCommand{\huge}{\@setfontsize\huge\@xxpt{28}}
257 \DeclareRobustCommand{\Huge}{\@setfontsize\Huge\@xxvpt{33}}
258 (/11pt)
259 (*12pt)
260 \DeclareRobustCommand{\scriptsize}{\@setfontsize\scriptsize\@viiipt{9.5}}
261 \DeclareRobustCommand{\tiny}{\@setfontsize\tiny\@vipt\@vipt}
262 \DeclareRobustCommand{\large}{\@setfontsize\large\@xivpt{21}}
263 \DeclareRobustCommand{\Large}{\@setfontsize\Large\@xxviipt{25}}
264 \DeclareRobustCommand{\LARGE}{\@setfontsize\LARGE\@xxpt{28}}
265 \DeclareRobustCommand{\huge}{\@setfontsize\huge\@xxvpt{33}}
266 \let\Huge=\huge
267 (/12pt)
268 (/10pt | 11pt | 12pt)
```

`\Cjascale` このクラスファイルが意図する和文スケール値 ($1\text{zw} \div \text{要求サイズ}$) を表す実数値
マクロ `\Cjascale` を定義します。この $\text{p}\text{I}\text{A}\text{T}\text{E}\text{X} 2_{\epsilon}$ の標準クラスでは、フォーマット
作成時に読み込まれたフォント定義ファイル (`jy1mc.fd` / `jy1gt.fd` / `jt1mc.fd` /
`jt1gt.fd`) での和文スケール値がそのまま有効ですので、これは 0.962216 です。

```
269 (*article | report | book)
270 \def\Cjascale{0.962216}
271 (/article | report | book)
```

26 レイアウト

26.1 用紙サイズの決定

`\columnsep` `\columnsep` は、二段組のときの、左右（あるいは上下）の段間の幅です。このス
`\columnseprule` ペースの中央に `\columnseprule` の幅の罫線が引かれます。

```
272 (*article | report | book)
273 \if@stysize
274 \tate \setlength\columnsep{3\Cwd}
275 \yoko \setlength\columnsep{2\Cwd}
276 \else
277 \setlength\columnsep{10\p@}
278 \fi
```

```
279 \setlength\columnseprule{0\p@}
```

26.2 段落の形

`\lineskip` これらの値は、行が近付き過ぎたときの $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ の動作を制御します。

```
\normallineskip280 \setlength\lineskip{1\p@}
281 \setlength\normallineskip{1\p@}
```

`\baselinestretch` これは、`\baselineskip` の倍率を示すために使います。デフォルトでは、**何もしません**。このコマンドが “empty” でない場合、`\baselineskip` の指定の `plus` や `minus` 部分は無視されることに注意してください。

```
282 \renewcommand{\baselinestretch}{}
```

`\parskip` `\parskip` は段落間に挿入される、縦方向の追加スペースです。`\parindent` は段落の先頭の字下げ幅です。

```
283 \setlength\parskip{0\p@ \@plus \p@}
284 \setlength\parindent{1\Cwd}
```

`\smallskipamount` これら 3 つのパラメータの値は、 $\mathrm{L}^{\mathrm{A}}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ カーネルの中で設定されています。これら
`\medskipamount` はおそらく、サイズオプションの指定によって変えるべきです。しかし、 $\mathrm{L}^{\mathrm{A}}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ 2.09
`\bigskipamount` や $\mathrm{L}^{\mathrm{A}}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ 2_ε の以前のリリースの両方との互換性を保つために、これらはまだ同じ値
としています。

```
285 (*10pt | 11pt | 12pt)
286 \setlength\smallskipamount{3\p@ \@plus 1\p@ \@minus 1\p@}
287 \setlength\medskipamount{6\p@ \@plus 2\p@ \@minus 2\p@}
288 \setlength\bigskipamount{12\p@ \@plus 4\p@ \@minus 4\p@}
289 (/10pt | 11pt | 12pt)
```

`\@lowpenalty` `\nopagebreak` と `\nolinebreak` コマンドは、これらのコマンドが置かれた場所に、
`\@medpenalty` ペナルティを起いて、分割を制御します。置かれるペナルティは、コマンドの引数に
`\@highpenalty` よって、`\@lowpenalty`, `\@medpenalty`, `\@highpenalty` のいずれかが使われます。

```
290 \@lowpenalty 51
291 \@medpenalty 151
292 \@highpenalty 301
293 (/article | report | book)
```

26.3 ページレイアウト

26.3.1 縦方向のスペース

`\headheight` `\headheight` は、ヘッダが入るボックスの高さです。`\headsep` は、ヘッダの下端
`\headsep` と本文領域との間の距離です。`\topskip` は、本文領域の上端と 1 行目のテキスト
`\topskip` のベースラインとの距離です。

```
294 (*10pt | 11pt | 12pt)
295 \setlength\headheight{12\p@}
296 (*tate)
297 \if@stysize
```



```

298 \ifnum\c@paper=2 % A5
299 \setlength\headsep{6mm}
300 \else % A4, B4, B5 and other
301 \setlength\headsep{8mm}
302 \fi
303 \else
304 \setlength\headsep{8mm}
305 \fi
306 \tate
307 *yoko
308 \bk\setlength\headsep{25\p@}
309 \10pt & bk\setlength\headsep{.25in}
310 \11pt & bk\setlength\headsep{.275in}
311 \12pt & bk\setlength\headsep{.275in}
312 \yoko
313 \setlength\topskip{1\ChT}

```

`\footskip` `\footskip` は、本文領域の下端とフッタの下端との距離です。フッタのボックスの高さを示す、`\footheight` は削除されました。

```

314 \tate\setlength\footskip{14mm}
315 *yoko
316 \bk\setlength\footskip{30\p@}
317 \10pt & bk\setlength\footskip{.35in}
318 \11pt & bk\setlength\footskip{.38in}
319 \12pt & bk\setlength\footskip{30\p@}
320 \yoko

```

`\maxdepth` $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ のプリミティブレジスタ `\maxdepth` は、`\topskip` と同じような働きをします。`\@maxdepth` レジスタは、つねに `\maxdepth` のコピーでなくてはなりません。これは `\begin{document}` の内部で設定されます。 $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ と $\mathrm{L}^{\mathrm{A}}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ 2.09 では、`\maxdepth` は 4pt に固定です。 $\mathrm{L}^{\mathrm{A}}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ 2 ϵ では、`\maxdepth+\topskip` を基本サイズの 1.5 倍にしたいので、`\maxdepth` を `\topskip` の半分の値で設定します。

```

321 \if@compatibility
322 \setlength\maxdepth{4\p@}
323 \else
324 \setlength\maxdepth{.5\topskip}
325 \fi

```

26.3.2 本文領域

`\textheight` と `\textwidth` は、本文領域の通常の高さと幅を示します。縦組でも横組でも、“高さ” は行数を、“幅” は字詰めを意味します。後ほど、これらの長さに `\topskip` の値が加えられます。

`\textwidth` 基本組の字詰めです。

互換モードの場合：

```
326 \if@compatibility
```

互換モード：a4j や b5j のクラスオプションが指定された場合の設定：

```

327 \if@stysize
328 \ifnum\c@@paper=2 % A5
329 \if@landscape
330 {10pt & yoko} \setlength\textwidth{47\Cwd}
331 {11pt & yoko} \setlength\textwidth{42\Cwd}
332 {12pt & yoko} \setlength\textwidth{40\Cwd}
333 {10pt & tate} \setlength\textwidth{27\Cwd}
334 {11pt & tate} \setlength\textwidth{25\Cwd}
335 {12pt & tate} \setlength\textwidth{23\Cwd}
336 \else
337 {10pt & yoko} \setlength\textwidth{28\Cwd}
338 {11pt & yoko} \setlength\textwidth{25\Cwd}
339 {12pt & yoko} \setlength\textwidth{24\Cwd}
340 {10pt & tate} \setlength\textwidth{46\Cwd}
341 {11pt & tate} \setlength\textwidth{42\Cwd}
342 {12pt & tate} \setlength\textwidth{38\Cwd}
343 \fi
344 \else\ifnum\c@@paper=3 % B4
345 \if@landscape
346 {10pt & yoko} \setlength\textwidth{75\Cwd}
347 {11pt & yoko} \setlength\textwidth{69\Cwd}
348 {12pt & yoko} \setlength\textwidth{63\Cwd}
349 {10pt & tate} \setlength\textwidth{53\Cwd}
350 {11pt & tate} \setlength\textwidth{49\Cwd}
351 {12pt & tate} \setlength\textwidth{44\Cwd}
352 \else
353 {10pt & yoko} \setlength\textwidth{60\Cwd}
354 {11pt & yoko} \setlength\textwidth{55\Cwd}
355 {12pt & yoko} \setlength\textwidth{50\Cwd}
356 {10pt & tate} \setlength\textwidth{85\Cwd}
357 {11pt & tate} \setlength\textwidth{76\Cwd}
358 {12pt & tate} \setlength\textwidth{69\Cwd}
359 \fi
360 \else\ifnum\c@@paper=4 % B5
361 \if@landscape
362 {10pt & yoko} \setlength\textwidth{60\Cwd}
363 {11pt & yoko} \setlength\textwidth{55\Cwd}
364 {12pt & yoko} \setlength\textwidth{50\Cwd}
365 {10pt & tate} \setlength\textwidth{34\Cwd}
366 {11pt & tate} \setlength\textwidth{31\Cwd}
367 {12pt & tate} \setlength\textwidth{28\Cwd}
368 \else
369 {10pt & yoko} \setlength\textwidth{37\Cwd}
370 {11pt & yoko} \setlength\textwidth{34\Cwd}
371 {12pt & yoko} \setlength\textwidth{31\Cwd}
372 {10pt & tate} \setlength\textwidth{55\Cwd}
373 {11pt & tate} \setlength\textwidth{51\Cwd}
374 {12pt & tate} \setlength\textwidth{47\Cwd}
375 \fi
376 \else % A4 ant other
377 \if@landscape
378 {10pt & yoko} \setlength\textwidth{73\Cwd}
379 {11pt & yoko} \setlength\textwidth{68\Cwd}
380 {12pt & yoko} \setlength\textwidth{61\Cwd}

```

```

381 <10pt & tate>      \setlength\textwidth{41\Cwd}
382 <11pt & tate>      \setlength\textwidth{38\Cwd}
383 <12pt & tate>      \setlength\textwidth{35\Cwd}
384     \else
385 <10pt & yoko>       \setlength\textwidth{47\Cwd}
386 <11pt & yoko>       \setlength\textwidth{43\Cwd}
387 <12pt & yoko>       \setlength\textwidth{40\Cwd}
388 <10pt & tate>       \setlength\textwidth{67\Cwd}
389 <11pt & tate>       \setlength\textwidth{61\Cwd}
390 <12pt & tate>       \setlength\textwidth{57\Cwd}
391     \fi
392     \fi\fi\fi
393     \else

```

互換モード：デフォルト設定

```

394     \if@twocolumn
395         \setlength\textwidth{52\Cwd}
396     \else
397 <10pt&!bk & yoko>    \setlength\textwidth{327\p@}
398 <11pt&!bk & yoko>    \setlength\textwidth{342\p@}
399 <12pt&!bk & yoko>    \setlength\textwidth{372\p@}
400 <10pt & bk & yoko>    \setlength\textwidth{4.3in}
401 <11pt & bk & yoko>    \setlength\textwidth{4.8in}
402 <12pt & bk & yoko>    \setlength\textwidth{4.8in}
403 <10pt & tate>        \setlength\textwidth{67\Cwd}
404 <11pt & tate>        \setlength\textwidth{61\Cwd}
405 <12pt & tate>        \setlength\textwidth{57\Cwd}
406     \fi
407     \fi

```

2e モードの場合：

```
408 \else
```

2e モード：a4j や b5j のクラスオプションが指定された場合の設定：二段組では用紙サイズの 8 割、一段組では用紙サイズの 7 割を版面の幅として設定します。

```

409     \if@stysize
410         \if@twocolumn
411 <yoko>              \setlength\textwidth{.8\paperwidth}
412 <tate>              \setlength\textwidth{.8\paperheight}
413     \else
414 <yoko>              \setlength\textwidth{.7\paperwidth}
415 <tate>              \setlength\textwidth{.7\paperheight}
416     \fi
417     \else

```

2e モード：デフォルト設定

```

418 <tate>              \setlength\@tempdima{\paperheight}
419 <yoko>              \setlength\@tempdima{\paperwidth}
420     \addtolength\@tempdima{-2in}
421 <tate>              \addtolength\@tempdima{-1.3in}
422 <yoko & 10pt>       \setlength\@tempdimb{327\p@}
423 <yoko & 11pt>       \setlength\@tempdimb{342\p@}
424 <yoko & 12pt>       \setlength\@tempdimb{372\p@}
425 <tate & 10pt>       \setlength\@tempdimb{67\Cwd}

```

```

426 <tate & 11pt>      \setlength\@tempdimb{61\Cwd}
427 <tate & 12pt>      \setlength\@tempdimb{57\Cwd}
428      \if@twocolumn
429          \ifdim\@tempdima>2\@tempdimb\relax
430              \setlength\textwidth{2\@tempdimb}
431          \else
432              \setlength\textwidth{\@tempdima}
433          \fi
434      \else
435          \ifdim\@tempdima>\@tempdimb\relax
436              \setlength\textwidth{\@tempdimb}
437          \else
438              \setlength\textwidth{\@tempdima}
439          \fi
440      \fi
441  \fi
442 \fi
443 \@settopoint\textwidth

```

\textheight 基本組の行数です。

互換モードの場合：

```

444 \if@compatibility

```

互換モード：a4j や b5j のクラスオプションが指定された場合の設定：

```

445     \if@stysize
446         \ifnum\c@@paper=2 % A5
447             \if@landscape
448 <10pt & yoko>      \setlength\textheight{17\Cvs}
449 <11pt & yoko>      \setlength\textheight{17\Cvs}
450 <12pt & yoko>      \setlength\textheight{16\Cvs}
451 <10pt & tate>      \setlength\textheight{26\Cvs}
452 <11pt & tate>      \setlength\textheight{26\Cvs}
453 <12pt & tate>      \setlength\textheight{25\Cvs}
454             \else
455 <10pt & yoko>      \setlength\textheight{28\Cvs}
456 <11pt & yoko>      \setlength\textheight{25\Cvs}
457 <12pt & yoko>      \setlength\textheight{24\Cvs}
458 <10pt & tate>      \setlength\textheight{16\Cvs}
459 <11pt & tate>      \setlength\textheight{16\Cvs}
460 <12pt & tate>      \setlength\textheight{15\Cvs}
461             \fi
462         \else\ifnum\c@@paper=3 % B4
463             \if@landscape
464 <10pt & yoko>      \setlength\textheight{38\Cvs}
465 <11pt & yoko>      \setlength\textheight{36\Cvs}
466 <12pt & yoko>      \setlength\textheight{34\Cvs}
467 <10pt & tate>      \setlength\textheight{48\Cvs}
468 <11pt & tate>      \setlength\textheight{48\Cvs}
469 <12pt & tate>      \setlength\textheight{45\Cvs}
470             \else
471 <10pt & yoko>      \setlength\textheight{57\Cvs}
472 <11pt & yoko>      \setlength\textheight{55\Cvs}
473 <12pt & yoko>      \setlength\textheight{52\Cvs}
474 <10pt & tate>      \setlength\textheight{33\Cvs}

```

```

475 {11pt & tate}          \setlength\textheight{33\Cvs}
476 {12pt & tate}          \setlength\textheight{31\Cvs}
477       \fi
478   \else\ifnum\c@@paper=4 % B5
479     \if@landscape
480 {10pt & yoko}          \setlength\textheight{22\Cvs}
481 {11pt & yoko}          \setlength\textheight{21\Cvs}
482 {12pt & yoko}          \setlength\textheight{20\Cvs}
483 {10pt & tate}          \setlength\textheight{34\Cvs}
484 {11pt & tate}          \setlength\textheight{34\Cvs}
485 {12pt & tate}          \setlength\textheight{32\Cvs}
486     \else
487 {10pt & yoko}          \setlength\textheight{35\Cvs}
488 {11pt & yoko}          \setlength\textheight{34\Cvs}
489 {12pt & yoko}          \setlength\textheight{32\Cvs}
490 {10pt & tate}          \setlength\textheight{21\Cvs}
491 {11pt & tate}          \setlength\textheight{21\Cvs}
492 {12pt & tate}          \setlength\textheight{20\Cvs}
493     \fi
494   \else % A4 and other
495     \if@landscape
496 {10pt & yoko}          \setlength\textheight{27\Cvs}
497 {11pt & yoko}          \setlength\textheight{26\Cvs}
498 {12pt & yoko}          \setlength\textheight{25\Cvs}
499 {10pt & tate}          \setlength\textheight{41\Cvs}
500 {11pt & tate}          \setlength\textheight{41\Cvs}
501 {12pt & tate}          \setlength\textheight{38\Cvs}
502     \else
503 {10pt & yoko}          \setlength\textheight{43\Cvs}
504 {11pt & yoko}          \setlength\textheight{42\Cvs}
505 {12pt & yoko}          \setlength\textheight{39\Cvs}
506 {10pt & tate}          \setlength\textheight{26\Cvs}
507 {11pt & tate}          \setlength\textheight{26\Cvs}
508 {12pt & tate}          \setlength\textheight{22\Cvs}
509     \fi
510   \fi\fi\fi
511 {yoko}          \addtolength\textheight{\topskip}
512 {bk & yoko}      \addtolength\textheight{\baselineskip}
513 {tate}          \addtolength\textheight{\Cht}
514 {tate}          \addtolength\textheight{\Cdp}

```

互換モード：デフォルト設定

```

515   \else
516 {10pt&!bk & yoko} \setlength\textheight{578\p@}
517 {10pt & bk & yoko} \setlength\textheight{554\p@}
518 {11pt & yoko}      \setlength\textheight{580.4\p@}
519 {12pt & yoko}      \setlength\textheight{586.5\p@}
520 {10pt & tate}      \setlength\textheight{26\Cvs}
521 {11pt & tate}      \setlength\textheight{25\Cvs}
522 {12pt & tate}      \setlength\textheight{24\Cvs}
523   \fi

```

2e モードの場合：

```

524 \else

```

2e モード: a4j や b5j のクラスオプションが指定された場合の設定: 縦組では用紙サイズの 70%(book) か 78%(article,report)、横組では 70%(book) か 75%(article,report) を版面の高さに設定します。

```

525 \if@stysize
526 <tate & bk> \setlength\textheight{.75\paperwidth}
527 <tate&!bk> \setlength\textheight{.78\paperwidth}
528 <yoko & bk> \setlength\textheight{.70\paperheight}
529 <yoko&!bk> \setlength\textheight{.75\paperheight}

```

2e モード: デフォルト値

```

530 \else
531 <tate> \setlength\@tempdima{\paperwidth}
532 <yoko> \setlength\@tempdima{\paperheight}
533 \addtolength\@tempdima{-2in}
534 <yoko> \addtolength\@tempdima{-1.5in}
535 \divide\@tempdima\baselineskip
536 \@tempcnta\@tempdima
537 \setlength\textheight{\@tempcnta\baselineskip}
538 \fi
539 \fi

```

最後に、\textheight に \topskip の値を加えます。

```

540 \addtolength\textheight{\topskip}
541 \@settopoint\textheight

```

26.3.3 マージン

\topmargin \topmargin は、“印字可能領域”—用紙の上端から 1 インチ内側—の上端からヘッダ部分の上端までの距離です。

2.09 互換モードの場合:

```

542 \if@compatibility
543 <*yoko>
544 \if@stysize
545 \setlength\topmargin{-.3in}
546 \else
547 <!bk> \setlength\topmargin{27\p@}
548 <10pt & bk> \setlength\topmargin{.75in}
549 <11pt & bk> \setlength\topmargin{.73in}
550 <12pt & bk> \setlength\topmargin{.73in}
551 \fi
552 </yoko>
553 <*tate>
554 \if@stysize
555 \ifnum\c@paper=2 % A5
556 \setlength\topmargin{.8in}
557 \else % A4, B4, B5 and other
558 \setlength\topmargin{32mm}
559 \fi
560 \else
561 \setlength\topmargin{32mm}
562 \fi
563 \addtolength\topmargin{-1in}

```

```

564 \addtolength\topmargin{-\headheight}
565 \addtolength\topmargin{-\headsep}
566 \tate
2e モードの場合：
567 \else
568 \setlength\topmargin{\paperheight}
569 \addtolength\topmargin{-\headheight}
570 \addtolength\topmargin{-\headsep}
571 \tate \addtolength\topmargin{-\textwidth}
572 \yoko \addtolength\topmargin{-\textheight}
573 \addtolength\topmargin{-\footskip}
574 \if@stysize
575 \ifnum\c@paper=2 % A5
576 \addtolength\topmargin{-1.3in}
577 \else
578 \addtolength\topmargin{-2.0in}
579 \fi
580 \else
581 \yoko \addtolength\topmargin{-2.0in}
582 \tate \addtolength\topmargin{-2.8in}
583 \fi
584 \addtolength\topmargin{-.5\topmargin}
585 \fi
586 \@settopoint\topmargin

```

`\marginparsep` `\marginparsep` は、本文と傍注の間にあけるスペースの幅です。横組では本文の左 `\marginparpush` (右) 端と傍注、縦組では本文の下 (上) 端と傍注の間になります。`\marginparpush` は、傍注と傍注との間のスペースの幅です。

```

587 \if@twocolumn
588 \setlength\marginparsep{10\p@}
589 \else
590 \tate \setlength\marginparsep{15\p@}
591 \yoko \setlength\marginparsep{10\p@}
592 \fi
593 \tate \setlength\marginparpush{7\p@}
594 \*yoko
595 \10pt \setlength\marginparpush{5\p@}
596 \11pt \setlength\marginparpush{5\p@}
597 \12pt \setlength\marginparpush{7\p@}
598 \yoko

```

`\oddsidemargin` まず、互換モードでの長さを示します。

`\evensidemargin` 互換モード、縦組の場合：

```

\marginparwidth599 \if@compatibility
600 \tate \setlength\oddsidemargin{0\p@}
601 \tate \setlength\evensidemargin{0\p@}
互換モード、横組、book クラスの場合：
602 \*yoko
603 \*bk

```

```

604 <10pt> \setlength\oddsidemargin {1.5in}
605 <11pt> \setlength\oddsidemargin {.25in}
606 <12pt> \setlength\oddsidemargin {.25in}
607 <10pt> \setlength\evensidemargin {1.5in}
608 <11pt> \setlength\evensidemargin {1.25in}
609 <12pt> \setlength\evensidemargin {1.25in}
610 <10pt> \setlength\marginparwidth {1.75in}
611 <11pt> \setlength\marginparwidth {1in}
612 <12pt> \setlength\marginparwidth {1in}
613 </bk>

```

互換モード、横組、report と article クラスの場合：

```

614 <*\bk>
615 \if@twoside
616 <10pt> \setlength\oddsidemargin {44\p@}
617 <11pt> \setlength\oddsidemargin {36\p@}
618 <12pt> \setlength\oddsidemargin {21\p@}
619 <10pt> \setlength\evensidemargin {82\p@}
620 <11pt> \setlength\evensidemargin {74\p@}
621 <12pt> \setlength\evensidemargin {59\p@}
622 <10pt> \setlength\marginparwidth {107\p@}
623 <11pt> \setlength\marginparwidth {100\p@}
624 <12pt> \setlength\marginparwidth {85\p@}
625 \else
626 <10pt> \setlength\oddsidemargin {60\p@}
627 <11pt> \setlength\oddsidemargin {54\p@}
628 <12pt> \setlength\oddsidemargin {39.5\p@}
629 <10pt> \setlength\evensidemargin {60\p@}
630 <11pt> \setlength\evensidemargin {54\p@}
631 <12pt> \setlength\evensidemargin {39.5\p@}
632 <10pt> \setlength\marginparwidth {90\p@}
633 <11pt> \setlength\marginparwidth {83\p@}
634 <12pt> \setlength\marginparwidth {68\p@}
635 \fi
636 </!bk>

```

互換モード、横組、二段組の場合：

```

637 \if@twocolumn
638 \setlength\oddsidemargin {30\p@}
639 \setlength\evensidemargin {30\p@}
640 \setlength\marginparwidth {48\p@}
641 \fi
642 </yoko>

```

縦組、横組にかかわらず、スタイルオプション設定ではゼロです。

```

643 \if@stysize
644 \if@twocolumn\else
645 \setlength\oddsidemargin{0\p@}
646 \setlength\evensidemargin{0\p@}
647 \fi
648 \fi

```

互換モードでない場合：

```

649 \else
650 \setlength\@tempdima{\paperwidth}

```



```

651 <tate> \addtolength\@tempdima{-\textheight}
652 <yoko> \addtolength\@tempdima{-\textwidth}

\oddsidemargin を計算します。
653 \if@twoside
654 <tate> \setlength\oddsidemargin{.6\@tempdima}
655 <yoko> \setlength\oddsidemargin{.4\@tempdima}
656 \else
657 \setlength\oddsidemargin{.5\@tempdima}
658 \fi
659 \addtolength\oddsidemargin{-1in}

\evensidemargin を計算します。
660 \setlength\evensidemargin{\paperwidth}
661 \addtolength\evensidemargin{-2in}
662 <tate> \addtolength\evensidemargin{-\textheight}
663 <yoko> \addtolength\evensidemargin{-\textwidth}
664 \addtolength\evensidemargin{-\oddsidemargin}
665 \@settopoint\oddsidemargin % 1999.1.6
666 \@settopoint\evensidemargin

\marginparwidth を計算します。ここで、\@tempdima の値は、
\paperwidth - \textwidth です。
667 <*yoko>
668 \if@twoside
669 \setlength\marginparwidth{.6\@tempdima}
670 \addtolength\marginparwidth{-.4in}
671 \else
672 \setlength\marginparwidth{.5\@tempdima}
673 \addtolength\marginparwidth{-.4in}
674 \fi
675 \ifdim \marginparwidth >2in
676 \setlength\marginparwidth{2in}
677 \fi
678 </yoko>

縦組の場合は、少し複雑です。
679 <*tate>
680 \setlength\@tempdima{\paperheight}
681 \addtolength\@tempdima{-\textwidth}
682 \addtolength\@tempdima{-\topmargin}
683 \addtolength\@tempdima{-\headheight}
684 \addtolength\@tempdima{-\headsep}
685 \addtolength\@tempdima{-\footskip}
686 \setlength\marginparwidth{.5\@tempdima}
687 </tate>
688 \@settopoint\marginparwidth
689 \fi

```

26.4 脚注

\footnotesep \footnotesep は、それぞれの脚注の先頭に置かれる“支柱”の高さです。このクラスでは、通常の \footnotesize の支柱と同じ長さですので、脚注間に余計な空白

は入りません。

```
690 \setlength\footnotesep{6.65\p@}
691 \setlength\footnotesep{7.7\p@}
692 \setlength\footnotesep{8.4\p@}
```

`\footins \skip\footins` は、本文の最終行と最初の脚注との間の距離です。

```
693 \setlength\skip\footins{9\p@ \@plus 4\p@ \@minus 2\p@}
694 \setlength\skip\footins{10\p@ \@plus 4\p@ \@minus 2\p@}
695 \setlength\skip\footins{10.8\p@ \@plus 4\p@ \@minus 2\p@}
```

26.5 フロート

すべてのフロートパラメータは、 \LaTeX のカーネルでデフォルトが定義されています。そのため、カウンタ以外のパラメータは `\renewcommand` で設定する必要があります。

26.5.1 フロートパラメータ

`\floatsep` フロートオブジェクトが本文のあるページに置かれるとき、フロートとそのページ
`\textfloatsep` にある別のオブジェクトの距離は、これらのパラメータで制御されます。これらの
`\intextsep` パラメータは、一段組モードと二段組モードの段抜きでないフロートの両方で使われます。

`\floatsep` は、ページ上部あるいは下部のフロート間の距離です。

`\textfloatsep` は、ページ上部あるいは下部のフロートと本文との距離です。

`\intextsep` は、本文の途中に出力されるフロートと本文との距離です。

```
696 (*10pt)
697 \setlength\floatsep {12\p@ \@plus 2\p@ \@minus 2\p@}
698 \setlength\textfloatsep{20\p@ \@plus 2\p@ \@minus 4\p@}
699 \setlength\intextsep {12\p@ \@plus 2\p@ \@minus 2\p@}
700 \/10pt)
701 (*11pt)
702 \setlength\floatsep {12\p@ \@plus 2\p@ \@minus 2\p@}
703 \setlength\textfloatsep{20\p@ \@plus 2\p@ \@minus 4\p@}
704 \setlength\intextsep {12\p@ \@plus 2\p@ \@minus 2\p@}
705 \/11pt)
706 (*12pt)
707 \setlength\floatsep {12\p@ \@plus 2\p@ \@minus 4\p@}
708 \setlength\textfloatsep{20\p@ \@plus 2\p@ \@minus 4\p@}
709 \setlength\intextsep {14\p@ \@plus 4\p@ \@minus 4\p@}
710 \/12pt)
```

`\dblfloatsep` 二段組モードで、`\textwidth` の幅を持つ、段抜きのフロートオブジェクトが本
`\dbltextfloatsep` 文と同じページに置かれるとき、本文とフロートとの距離は、`\dblfloatsep` と
`\dbltextfloatsep` によって制御されます。

`\dblfloatsep` は、ページ上部あるいは下部のフロートと本文との距離です。

`\dbltextfloatsep` は、ページ上部あるいは下部のフロート間の距離です。

```
711 (*10pt)
```

```

712 \setlength\dblfloatsep {12\p@ \@plus 2\p@ \@minus 2\p@}
713 \setlength\dbltextfloatsep{20\p@ \@plus 2\p@ \@minus 4\p@}
714 </10pt>
715 < *11pt>
716 \setlength\dblfloatsep {12\p@ \@plus 2\p@ \@minus 2\p@}
717 \setlength\dbltextfloatsep{20\p@ \@plus 2\p@ \@minus 4\p@}
718 </11pt>
719 < *12pt>
720 \setlength\dblfloatsep {14\p@ \@plus 2\p@ \@minus 4\p@}
721 \setlength\dbltextfloatsep{20\p@ \@plus 2\p@ \@minus 4\p@}
722 </12pt>

```

`\@fptop` フロートオブジェクトが、独立したページに置かれるとき、このページのレイアウトは、次のパラメータで制御されます。これらのパラメータは、一段組モードか、`\@fpbot` 二段組モードでの一段出力のフロートオブジェクトに対して使われます。

ページ上部では、`\@fptop` の伸縮長が挿入されます。ページ下部では、`\@fpbot` の伸縮長が挿入されます。フロート間には `\@fpsep` が挿入されます。

なお、そのページを空白で満たすために、`\@fptop` と `\@fpbot` の少なくともどちらか一方に、`plus ...fil` を含めてください。

```

723 < *10pt>
724 \setlength\@fptop{0\p@ \@plus 1fil}
725 \setlength\@fpsep{8\p@ \@plus 2fil}
726 \setlength\@fpbot{0\p@ \@plus 1fil}
727 </10pt>
728 < *11pt>
729 \setlength\@fptop{0\p@ \@plus 1fil}
730 \setlength\@fpsep{8\p@ \@plus 2fil}
731 \setlength\@fpbot{0\p@ \@plus 1fil}
732 </11pt>
733 < *12pt>
734 \setlength\@fptop{0\p@ \@plus 1fil}
735 \setlength\@fpsep{10\p@ \@plus 2fil}
736 \setlength\@fpbot{0\p@ \@plus 1fil}
737 </12pt>

```

`\@dblftop` 二段組モードでの二段抜きのフロートに対しては、これらのパラメータが使われ
`\@dblfpsep` ます。

```

\@dblfpbot 738 < *10pt>
739 \setlength\@dblftop{0\p@ \@plus 1fil}
740 \setlength\@dblfpsep{8\p@ \@plus 2fil}
741 \setlength\@dblfpbot{0\p@ \@plus 1fil}
742 </10pt>
743 < *11pt>
744 \setlength\@dblftop{0\p@ \@plus 1fil}
745 \setlength\@dblfpsep{8\p@ \@plus 2fil}
746 \setlength\@dblfpbot{0\p@ \@plus 1fil}
747 </11pt>
748 < *12pt>
749 \setlength\@dblftop{0\p@ \@plus 1fil}
750 \setlength\@dblfpsep{10\p@ \@plus 2fil}
751 \setlength\@dblfpbot{0\p@ \@plus 1fil}

```

752 $\langle /12pt \rangle$
753 $\langle /10pt | 11pt | 12pt \rangle$

26.5.2 フロートオブジェクトの上限値

`\c@topnumber` *topnumber* は、本文ページの上部に出力できるフロートの最大数です。

754 $\langle *article | report | book \rangle$
755 `\setcounter{topnumber}{2}`

`\c@bottomnumber` *bottomnumber* は、本文ページの下部に出力できるフロートの最大数です。

756 `\setcounter{bottomnumber}{1}`

`\c@totalnumber` *totalnumber* は、本文ページに出力できるフロートの最大数です。

757 `\setcounter{totalnumber}{3}`

`\c@dbltopnumber` *dbltopnumber* は、二段組時における、本文ページの上部に出力できる段抜きのフロートの最大数です。

758 `\setcounter{dbltopnumber}{2}`

`\topfraction` これは、本文ページの上部に出力されるフロートが占有できる最大の割合です。

759 `\renewcommand{\topfraction}{.7}`

`\bottomfraction` これは、本文ページの下部に出力されるフロートが占有できる最大の割合です。

760 `\renewcommand{\bottomfraction}{.3}`

`\textfraction` これは、本文ページに最低限、入らなくてはならない本文の割合です。

761 `\renewcommand{\textfraction}{.2}`

`\floatpagefraction` これは、フロートだけのページで最低限、入らなくてはならないフロートの割合です。

762 `\renewcommand{\floatpagefraction}{.5}`

`\dbltopfraction` これは、2 段組時における本文ページに、2 段抜きのフロートが占めることができる最大の割合です。

763 `\renewcommand{\dbltopfraction}{.7}`

`\dblfloatpagefraction` これは、2 段組時におけるフロートだけのページに最低限、入らなくてはならない 2 段抜きのフロートの割合です。

764 `\renewcommand{\dblfloatpagefraction}{.5}`

27 改ページ（日本語 T_EX 開発コミュニティ版のみ）

`\pltx@cleartorightpage` `\cleardoublepage` 命令は、L^AT_EX カーネルでは「奇数ページになるまでページを
`\pltx@cleartoleftpage` 繰る命令」として定義されています。しかし pL^AT_EX カーネルでは、アスキーの方
`\pltx@cleartooddpage` 針により「横組では奇数ページになるまで、縦組では偶数ページになるまでページ
`\pltx@cleartoevenpage`

を繰る命令」に再定義されています。すなわち、pL^AT_EX では縦組でも横組でも右ページになるまでページを繰ることになります。

pL^AT_EX 標準クラスの book は、横組も縦組も openright がデフォルトになっていて、これは従来 pL^AT_EX カーネルで定義された `\cleardoublepage` を利用していました。しかし、縦組で奇数ページ始まりの文書を作りたい場合もあるでしょうから、コミュニティ版クラスでは以下の（非ユーザ向け）命令を追加します。

1. `\pltx@cleartorightpage`：右ページになるまでページを繰る命令
2. `\pltx@cleartoleftpage`：左ページになるまでページを繰る命令
3. `\pltx@cleartooddpage`：奇数ページになるまでページを繰る命令
4. `\pltx@cleartoevenpage`：偶数ページになるまでページを繰る命令

```
765 \def\pltx@cleartorightpage{\clearpage\if@twoside
766   \ifodd\c@page
767     \iftdir
768       \hbox{}\thispagestyle{empty}\newpage
769       \if@twocolumn\hbox{}\newpage\fi
770     \fi
771   \else
772     \ifydir
773       \hbox{}\thispagestyle{empty}\newpage
774       \if@twocolumn\hbox{}\newpage\fi
775     \fi
776   \fi\fi}
777 \def\pltx@cleartoleftpage{\clearpage\if@twoside
778   \ifodd\c@page
779     \ifydir
780       \hbox{}\thispagestyle{empty}\newpage
781       \if@twocolumn\hbox{}\newpage\fi
782     \fi
783   \else
784     \iftdir
785       \hbox{}\thispagestyle{empty}\newpage
786       \if@twocolumn\hbox{}\newpage\fi
787     \fi
788   \fi\fi}
```

`\pltx@cleartooddpage` は L^AT_EX の `\cleardoublepage` に似ていますが、上の 2 つに合わせるため `\thispagestyle{empty}` を追加してあります。

```
789 \def\pltx@cleartooddpage{\clearpage\if@twoside
790   \ifodd\c@page\else
791     \hbox{}\thispagestyle{empty}\newpage
792     \if@twocolumn\hbox{}\newpage\fi
793   \fi\fi}
794 \def\pltx@cleartoevenpage{\clearpage\if@twoside
795   \ifodd\c@page
796     \hbox{}\thispagestyle{empty}\newpage
797     \if@twocolumn\hbox{}\newpage\fi
798   \fi\fi}
```

`\cleardoublepage` そして report と book クラスの場合は、ユーザ向け命令である `\cleardoublepage` を、`openright` オプションが指定されている場合は `\pltx@cleartorightpage` に、`openleft` オプションが指定されている場合は `\pltx@cleartoleftpage` に、それぞれ `\let` します。openany の場合は pL^AT_EX カーネルの定義のままです。

```

799 (*!article)
800 \if@openleft
801   \let\cleardoublepage\pltx@cleartoleftpage
802 \else\if@openright
803   \let\cleardoublepage\pltx@cleartorightpage
804 \fi\fi
805 \!/article)

```

28 ページスタイル

pL^AT_EX 2_ε では、つぎの6種類のページスタイルを使用できます。`empty` は `ltpage.dtx` で定義されています。

<code>empty</code>	ヘッダにもフッタにも出力しない
<code>plain</code>	フッタにページ番号のみを出力する
<code>headnombre</code>	ヘッダにページ番号のみを出力する
<code>footnombre</code>	フッタにページ番号のみを出力する
<code>headings</code>	ヘッダに見出しとページ番号を出力する
<code>bothstyle</code>	ヘッダに見出し、フッタにページ番号を出力する

ページスタイル `foo` は、`\ps@foo` コマンドとして定義されます。

`\@evenhead` これらは `\ps@...` から呼び出され、ヘッダとフッタを出力するマクロです。

<code>\@oddhead</code>	<code>\@oddhead</code>	奇数ページのヘッダを出力
<code>\@evenfoot</code>	<code>\@oddfoot</code>	奇数ページのフッタを出力
<code>\@oddfoot</code>	<code>\@evenhead</code>	偶数ページのヘッダを出力
	<code>\@evenfoot</code>	偶数ページのフッタを出力

これらの内容は、横組の場合は `\textwidth` の幅を持つ `\hbox` に入れられ、縦組の場合は `\textheight` の幅を持つ `\hbox` に入れられます。

28.1 マークについて

ヘッダに入る章番号や章見出しは、見出しコマンドで実行されるマークコマンドで決定されます。ここでは、実行されるマークコマンドの定義を行なっています。これらのマークコマンドは、T_EX の `\mark` 機能を用いて、‘left’ と ‘right’ の2種類のマークを生成するように定義しています。

`\markboth{<LEFT>}{<RIGHT>}`: 両方のマークに追加します。

`\markright{<RIGHT>}`: ‘右’ マークに追加します。

`\leftmark`: `\@oddhead`, `\@oddfoot`, `\@evenhead`, `\@evenfoot` マクロで使われ、現在の“左”マークを出力します。`\leftmark` は T_EX の `\botmark` コマンドのような働きをします。初期値は空でなくてはいけません。

`\rightmark`: `\@oddhead`, `\@oddfoot`, `\@evenhead`, `\@evenfoot` マクロで使われ、現在の“右”マークを出力します。`\rightmark` は $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ の `\firstmark` コマンドのような働きをします。初期値は空でなくてははいけません。

マークコマンドの動作は、左マークの‘範囲内の’右マークのために合理的になっています。たとえば、左マークは `\chapter` コマンドによって変更されます。そして右マークは `\section` コマンドによって変更されます。しかし、同一ページに複数の `\markboth` コマンドが現れたとき、おかしい結果となることがあります。

`\tableofcontents` のようなコマンドは、`\@mkboth` コマンドを用いて、あるページスタイルの中でマークを設定しなくてはなりません。`\@mkboth` は、`\ps@...` コマンドによって、`\markboth` (ヘッダを設定する) か、`\@gobbletwo` (何もしない) に `\let` されます。

28.2 plain ページスタイル

`\ps@plain jpl@in` に `\let` するために、ここで定義をします。

```
806 \def\ps@plain{\let\@mkboth\@gobbletwo
807   \let\ps@jpl@in\ps@plain
808   \let\@oddhead\@empty
809   \def\@oddfoot{\reset@font\hfil\thepage\hfil}%
810   \let\@evenhead\@empty
811   \let\@evenfoot\@oddfoot}
```

28.3 jpl@in ページスタイル

`\ps@jpl@in jpl@in` スタイルは、クラスファイル内部で使用するものです。 $\mathrm{L}^{\mathrm{A}}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ では、book クラスを *headings* としています。しかし、`\tableofcontents` コマンドの内部では *plain* として設定されるため、一つの文書でのページ番号の位置が上下に出力されることになります。

そこで、 $\mathrm{pL}^{\mathrm{A}}\mathrm{T}_{\mathrm{E}}\mathrm{X} 2_{\epsilon}$ では、`\tableofcontents` や `\theindex` のページスタイルを *jpl@in* にし、実際に出力される形式は、ほかのページスタイルで `\let` をしています。したがって、*headings* のとき、目次ページのページ番号はヘッダ位置に出力され、*plain* のときには、フッタ位置に出力されます。

ここで、定義をしているのは、その初期値です。

```
812 \let\ps@jpl@in\ps@plain
```

28.4 headnombre ページスタイル

`\ps@headnombre headnombre` スタイルは、ヘッダにページ番号のみを出力します。

```
813 \def\ps@headnombre{\let\@mkboth\@gobbletwo
814   \let\ps@jpl@in\ps@headnombre
815   \yoko \def\@evenhead{\thepage\hfil}%
816   \yoko \def\@oddhead{\hfil\thepage}%
817   \tate \def\@evenhead{\hfil\thepage}%
818   \tate \def\@oddhead{\thepage\hfil}%
819   \let\@oddfoot\@empty\let\@evenfoot\@empty}
```

28.5 footnombre ページスタイル

`\ps@footnombre` *footnombre* スタイルは、フッタにページ番号のみを出力します。

```
820 \def\ps@footnombre{\let\mkboth\gobbletwo
821   \let\ps@jpl@in\ps@footnombre
822   \def\@evenfoot{\thepage\hfil}%
823   \def\@oddfoot{\hfil\thepage}%
824   \def\@evenfoot{\hfil\thepage}%
825   \def\@oddfoot{\thepage\hfil}%
826   \let\@oddhead\empty\let\@evenhead\empty}
```

28.6 headings スタイル

headings スタイルは、ヘッダに見出しとページ番号を出力します。

`\ps@headings` このスタイルは、両面印刷と片面印刷とで形式が異なります。

```
827 \if@twoside
```

横組の場合は、奇数ページが右に、偶数ページが左にきます。縦組の場合は、奇数ページが左に、偶数ページが右にきます。

```
828 \def\ps@headings{\let\ps@jpl@in\ps@headnombre
829   \let\@oddfoot\empty\let\@evenfoot\empty
830   \def\@evenhead{\thepage\hfil\leftmark}%
831   \def\@oddhead{\rightmark\hfil\thepage}%
832   \def\@evenhead{\leftmark\hfil\thepage}%
833   \def\@oddhead{\thepage\hfil\rightmark}%
834   \let\mkboth\markboth
835   (*article)
836   \def\sectionmark##1{\markboth{%
837     \ifnum \c@secnumdepth >\z@ \thesection.\hskip1zw\fi
838     ##1}{}}%
839   \def\subsectionmark##1{\markright{%
840     \ifnum \c@secnumdepth >\@ne \thesubsection.\hskip1zw\fi
841     ##1}}%
842   (/article)
843   (*report | book)
844   \def\chaptermark##1{\markboth{%
845     \ifnum \c@secnumdepth >\m@ne
846     \if@mainmatter
847       \@chapapp\thechapter\@chappos\hskip1zw
848     \if
849       \fi
850     ##1}{}}%
851   \def\sectionmark##1{\markright{%
852     \ifnum \c@secnumdepth >\z@ \thesection.\hskip1zw\fi
853     ##1}}%
854   (/report | book)
855   }
```

片面印刷の場合：

```
856 \else % if not twoside
857   \def\ps@headings{\let\ps@jpl@in\ps@headnombre
858     \let\@oddfoot\empty
```



```

859 <yoko> \def\@oddhead{\rightmark\hfil\thepage}%
860 <tate> \def\@oddhead{\thepage\hfil\rightmark}%
861 \let\@mkboth\markboth
862 <*article>
863 \def\sectionmark##1{\markright{%
864 \ifnum \c@secnumdepth >\m@ne \thesection.\hskip1zw\fi
865 ##1}}%
866 </article>
867 <*report | book>
868 \def\chaptermark##1{\markright{%
869 \ifnum \c@secnumdepth >\m@ne
870 <book> \if@mainmatter
871 \chapapp\thechapter\@chappos\hskip1zw
872 <book> \fi
873 \fi
874 ##1}}%
875 </report | book>
876 }
877 \fi

```

28.7 bothstyle スタイル

`\ps@bothstyle` *bothstyle* スタイルは、ヘッダに見出しを、フッタにページ番号を出力します。

このスタイルは、両面印刷と片面印刷とで形式が異なります。

```

878 \if@twoside
879 \def\ps@bothstyle{\let\ps@jpl@in\ps@footnombre
880 <*yoko>
881 \def\@evenhead{\leftmark\hfil}% right page
882 \def\@evenfoot{\thepage\hfil}% right page
883 \def\@oddhead{\hfil\rightmark}% left page
884 \def\@oddfoot{\hfil\thepage}% left page
885 </yoko>
886 <*tate>
887 \def\@evenhead{\hfil\leftmark}% right page
888 \def\@evenfoot{\hfil\thepage}% right page
889 \def\@oddhead{\rightmark\hfil}% left page
890 \def\@oddfoot{\thepage\hfil}% left page
891 </tate>
892 \let\@mkboth\markboth
893 <*article>
894 \def\sectionmark##1{\markboth{%
895 \ifnum \c@secnumdepth >\z@ \thesection.\hskip1zw\fi
896 ##1}{}}%
897 \def\subsectionmark##1{\markright{%
898 \ifnum \c@secnumdepth >\@ne \thesubsection.\hskip1zw\fi
899 ##1}}%
900 </article>
901 <*report | book>
902 \def\chaptermark##1{\markboth{%
903 \ifnum \c@secnumdepth >\m@ne
904 <book> \if@mainmatter
905 \chapapp\thechapter\@chappos\hskip1zw
906 <book> \fi

```

```

907 \fi
908 ##1}{}}%
909 \def\sectionmark##1{\markright{%
910 \ifnum \c@secnumdepth >\z@ \thesection.\hskip1zw\fi
911 ##1}}%
912 </report | book>
913 }

914 \else % if one column
915 \def\ps@bothstyle{\let\ps@jpl@in\ps@footnombre
916 \yoko \def\@oddhead{\hfil\rightmark}%
917 \yoko \def\@oddfoot{\hfil\thepage}%
918 \tate \def\@oddhead{\rightmark\hfil}%
919 \tate \def\@oddfoot{\thepage\hfil}%
920 \let\@mkboth\markboth
921 <*article>
922 \def\sectionmark##1{\markright{%
923 \ifnum \c@secnumdepth >\m@ne \thesection.\hskip1zw\fi
924 ##1}}%
925 </article>
926 <*report | book>
927 \def\chaptermark##1{\markright{%
928 \ifnum \c@secnumdepth >\m@ne
929 \book \if@mainmatter
930 \@chapapp\thechapter\@chappos\hskip1zw
931 \book \fi
932 \fi
933 ##1}}%
934 </report | book>
935 }
936 \fi

```

28.8 myheading スタイル

`\ps@myheadings` *myheadings* ページスタイルは簡潔に定義されています。ユーザがページスタイルを設計するときのヒナ型として使用することができます。

```

937 \def\ps@myheadings{\let\ps@jpl@in\ps@plain%
938 \let\@oddfoot\@empty\let\@evenfoot\@empty
939 \yoko \def\@evenhead{\thepage\hfil\leftmark}%
940 \yoko \def\@oddhead{\rightmark\hfil\thepage}%
941 \tate \def\@evenhead{\leftmark\hfil\thepage}%
942 \tate \def\@oddhead{\thepage\hfil\rightmark}%
943 \let\@mkboth\@gobbletwo
944 \!article \let\chaptermark\@gobble
945 \let\sectionmark\@gobble
946 \article \let\subsectionmark\@gobble
947 }

```

29 文書コマンド

29.1 表題

`\title` 文書のタイトル、著者、日付の情報のための、これらの3つのコマンドは `ltsect.dtx` `\author` で提供されています。これらのコマンドは次のように定義されています。

```
\date 948 %\DeclareRobustCommand*\title}[1]{\gdef\@title{#1}}
949 %\DeclareRobustCommand*\author}[1]{\gdef\@author{#1}}
950 %\DeclareRobustCommand*\date}[1]{\gdef\@date{#1}}
```

`\date` マクロのデフォルトは、今日の日付です。

```
951 %\date{\today}
```

`titlepage` (*env.*) 通常環境では、ページの最初と最後を除き、タイトルページ環境は何もしません。また、ページ番号の出力を抑制します。レポートスタイルでは、ページ番号を1にリセットし、そして最後で1に戻します。互換モードでは、ページ番号はゼロに設定されますが、右起しページ用のページパラメータでは誤った結果になります。二段組スタイルでも一段組のページが作られます。

日本語 $T_E X$ 開発コミュニティによる変更: 上にあるのはアスキー版の説明です。改めてアスキー版の挙動を整理すると、以下のようになります。

1. アスキー版では、タイトルページの番号を必ず1にリセットしていましたが、これは正しくありません。これは、タイトルページが奇数ページ目か偶数ページ目にかかわらず、レイアウトだけ奇数ページ用が適用されてしまうからです。さらに、タイトルの次のページも偶数のページ番号を持つってしまうため、両面印刷で奇数ページと偶数ページが交互に出なくなるという問題もあります。
2. アスキー版 `book` クラスは、タイトルページを必ず `\cleardoublepage` で始めていました。pL $T_E X$ カーネルでの `\cleardoublepage` の定義から、縦組の既定ではタイトルが偶数ページ目に出ることになります。これ自体が正しくないと断定することはできませんが、タイトルのページ番号を1にリセットすることと合わさって、偶数ページに送ったタイトルに奇数ページ用レイアウトが適用されてしまうという結果は正しくありません。

そこで、コミュニティ版ではタイトルのレイアウトが必ず奇数ページ用になるという挙動を支持し、`book` クラスではタイトルページを奇数ページ目に送ることにしました。これでタイトルページが表紙らしく見えるようになります。また、`report` クラスのようなタイトルが成り行きに従って出る場合には

- 奇数ページ目に出る場合、ページ番号を1（奇数）にリセット
- 偶数ページ目に出る場合、ページ番号を0（偶数）にリセット

としました。

一つめの例を考えます。

```
\documentclass{tbook}
\title{タイトル}\author{著者}
\begin{document}
\maketitle
\chapter{チャプター}
\end{document}
```

アスキー版 tbook クラスでの結果は

- 1 ページ目：空白（ページ番号 1 は非表示）
- 2 ページ目：タイトル（奇数レイアウト、ページ番号 1 は非表示）
- 3 ページ目：チャプター（偶数レイアウト、ページ番号 2）

ですが、仮に最初の空白ページさえなければ

- 1 ページ目：タイトルすなわち表紙（奇数レイアウト、ページ番号 1 は非表示）
- 2 ページ目：チャプター（偶数レイアウト、ページ番号 2）

とみなせるため、コミュニティ版では空白ページを発生させないようにしました。

二つめの例を考えます。

```
\documentclass{tbook}
\title{タイトル}\author{著者}
\begin{document}
テスト文章
\maketitle
\chapter{チャプター}
\end{document}
```

アスキー版 tbook クラスでの結果は

- 1 ページ目：テスト文章（奇数レイアウト、ページ番号 1）
- 2 ページ目：タイトル（奇数レイアウト、ページ番号 1 は非表示）
- 3 ページ目：チャプター（偶数レイアウト、ページ番号 2）

ですが、これでは奇数と偶数のページ番号が交互になっていないので正しくありません。そこで、コミュニティ版では

- 1 ページ目：テスト文章（奇数レイアウト、ページ番号 1）
- 2 ページ目：空白ページ（ページ番号 2 は非表示）
- 3 ページ目：タイトル（奇数レイアウト、ページ番号 1 は非表示）
- 4 ページ目：チャプター（偶数レイアウト、ページ番号 2）

と直しました。

なお、pL^AT_EX 2.09 互換モードはアスキー版のまま、すなわち「ページ番号をゼロに設定」としてあります。これは、横組の右起こしの挙動としては誤りですが、縦組の右起こしの挙動としては一応正しくなっているといえます。

最初に互換モードの定義を作ります。

```
952 \if@compatibility
953 \newenvironment{titlepage}
954     {%
955 \book} \cleardoublepage
```

```

956     \if@twocolumn\@restonecoltrue\onecolumn
957     \else\@restonecolfalse\newpage\fi
958     \thispagestyle{empty}%
959     \setcounter{page}\z@
960 }%
961 {\if@restonecol\twocolumn\else\newpage\fi
962 }

```

そして、 \LaTeX ネイティブのための定義です。

```

963 \else
964 \newenvironment{titlepage}
965 {%
966 <book>      \pltx@cleartooddpage %% 2017/02/15
967     \if@twocolumn
968     \@restonecoltrue\onecolumn
969     \else
970     \@restonecolfalse\newpage
971     \fi
972     \thispagestyle{empty}%
973     \ifodd\c@page\setcounter{page}\@ne\else\setcounter{page}\z@\fi %% 2017/02/15
974 }%
975 {\if@restonecol\twocolumn \else \newpage \fi

```

両面モードでなければ、タイトルページの直後のページのページ番号も 1 にします。

```

976     \if@twoside\else
977     \setcounter{page}\@ne
978     \fi
979 }
980 \fi

```

`\maketitle` このコマンドは、表題を作成し、出力します。表題ページを独立させるかどうかによって定義が異なります。report と book クラスのデフォルトは独立した表題です。article クラスはオプションで独立させることができます。

`\p@thanks` 縦組のときは、`\thanks` コマンドを `\p@thanks` に `\let` します。このコマンドは `\footnotetext` を使わず、直接、文字を `\@thanks` に格納していきます。

著者名の脇に表示される合印は直立した数字、注釈側は横に寝た数字となっていました。不自然なので `\hbox{\yoko ...}` を追加し、両方とも直立するようにしました。

```

981 \def\p@thanks#1{\footnotemark
982   \protected@xdef\@thanks{\@thanks
983     \protect{\noindent\hbox{\yoko$\m@th^{\thefootnote$}\#1\protect\par}}}

984 \if@titlepage
985   \newcommand{\maketitle}{\begin{titlepage}%
986     \let\footnotesize\small
987     \let\footnoterule\relax
988     \tate \let\thanks\p@thanks
989     \let\footnote\thanks

990     \vbox to\textheight\bgroup\tate\hsize\textwidth
991     \null\vfil

```

```

992 \vskip 60\p@
993 \begin{center}%
994   {\LARGE \@title \par}%
995   \vskip 3em%
996   {\Large
997     \lineskip .75em%
998     \begin{tabular}[t]{c}%
999       \@author
1000     \end{tabular}\par}%
1001   \vskip 1.5em%
1002   {\large \@date \par}%      % Set date in \large size.
1003 \end{center}\par
1004 \tate \vfil{\centering\@thanks}\vfil\null
1005 \tate \egroup
1006 \yoko \@thanks\vfil\null
1007 \end{titlepage}%

```

footnote カウンタをリセットし、\thanks と \maketitle コマンドを無効にし、いくつかの内部マクロを空にして格納領域を節約します。

```

1008 \setcounter{footnote}{0}%
1009 \global\let\thanks\relax
1010 \global\let\maketitle\relax
1011 \global\let\p@thanks\relax
1012 \global\let\@thanks\@empty
1013 \global\let\@author\@empty
1014 \global\let\@date\@empty
1015 \global\let\@title\@empty

```

タイトルが組版されたら、\title コマンドなどの宣言を無効にできます。 \and の定義は、\author の引数でのみ使用しますので、破棄します。

```

1016 \global\let\title\relax
1017 \global\let\author\relax
1018 \global\let\date\relax
1019 \global\let\and\relax
1020 }%
1021 \else
1022 \newcommand{\maketitle}{\par
1023 \begingroup
1024   \renewcommand{\thefootnote}{\fnsymbol{footnote}}%
1025   \def\@makefnmark{\hbox{\ifdir $ \m@th^{\@thefnmark}$
1026     \else\hbox{\yoko$\m@th^{\@thefnmark}$}\fi}}%
1027 \tate
1028   \long\def\@makefntext##1{\parindent 1zw\noindent
1029     \hb@xt@ 2zw{\hss\@makefnmark}##1}%
1030 \tate
1031 \yoko
1032   \long\def\@makefntext##1{\parindent 1em\noindent
1033     \hb@xt@ 1.8em{\hss$\m@th^{\@thefnmark}$}##1}%
1034 \yoko
1035   \if@twocolumn
1036     \ifnum \col@number=\@one \maketitle
1037   \else \twocolumn[\maketitle]%
1038   \fi
1039 \else

```

```

1040 \newpage
1041 \global\@topnum\z@ % Prevents figures from going at top of page.
1042 \@maketitle
1043 \fi
1044 \thispagestyle{jpl@in}\@thanks

```

ここでグループを閉じ、*footnote* カウンタをリセットし、`\thanks`、`\maketitle`、`\@maketitle` を無効にし、いくつかの内部マクロを空にして格納領域を節約します。

```

1045 \endgroup
1046 \setcounter{footnote}{0}%
1047 \global\let\thanks\relax
1048 \global\let\maketitle\relax
1049 \global\let\@maketitle\relax
1050 \global\let\p@thanks\relax
1051 \global\let\@thanks\@empty
1052 \global\let\@author\@empty
1053 \global\let\@date\@empty
1054 \global\let\@title\@empty
1055 \global\let\ttitle\relax
1056 \global\let\author\relax
1057 \global\let\date\relax
1058 \global\let\and\relax
1059 }

```

`\@maketitle` 独立した表題ページを作らない場合の、表題の出力形式です。

```

1060 \def\@maketitle{%
1061 \newpage\null
1062 \vskip 2em%
1063 \begin{center}%
1064 (yoko) \let\footnote\thanks
1065 (tate) \let\footnote\p@thanks
1066 {\LARGE \@title \par}%
1067 \vskip 1.5em%
1068 {\large
1069 \lineskip .5em%
1070 \begin{tabular}[t]{c}%
1071 \@author
1072 \end{tabular}\par}%
1073 \vskip 1em%
1074 {\large \@date}%
1075 \end{center}%
1076 \par\vskip 1.5em}
1077 \fi

```

29.2 概要

`abstract (env.)` 要約文のための環境です。book クラスでは使えません。report スタイルと、`titlepage` オプションを指定した article スタイルでは、独立したページに出力されます。

```

1078 (*article|report)
1079 \if@titlepage
1080 \newenvironment{abstract}{%

```

```

1081 \titlepage
1082 \null\vfil
1083 \@beginparpenalty\@lowpenalty
1084 \begin{center}%
1085   {\bfseries\abstractname}%
1086   \@endparpenalty\@M
1087 \end{center}}%
1088 {\par\vfil\null\endtitlepage}
1089 \else
1090 \newenvironment{abstract}{%
1091   \if@twocolumn
1092     \section*{\abstractname}%
1093   \else
1094     \small
1095     \begin{center}%
1096       {\bfseries\abstractname\vspace{-.5em}\vspace{\z@}}%
1097     \end{center}%
1098     \quotation
1099   \fi}{\if@twocolumn\else\endquotation\fi}
1100 \fi
1101 </article | report>

```

29.3 章見出し

29.3.1 マークコマンド

`\chaptermark` `\...mark` コマンドを初期化します。これらのコマンドはページスタイルの定義で `\sectionmark` 使われます (第 28 節参照)。これらのたいていのコマンドは `ltsect.dtx` ですでに `\subsectionmark` 定義されています。

```

\subsubsectionmark 102 \!article\newcommand*\chaptermark}[1]{%
\paragraphmark 1103 %\newcommand*\sectionmark}[1]{%
\subparagraphmark 1104 %\newcommand*\subsectionmark}[1]{%
1105 %\newcommand*\subsubsectionmark}[1]{%
1106 %\newcommand*\paragraphmark}[1]{%
1107 %\newcommand*\subparagraphmark}[1]{%

```

29.3.2 カウンタの定義

`\c@secnumdepth` `secnumdepth` には、番号を付ける、見出しコマンドのレベルを設定します。

```

1108 <article>\setcounter{secnumdepth}{3}
1109 \!article\setcounter{secnumdepth}{2}

```

`\c@chapter` これらのカウンタは見出し番号に使われます。最初の引数は、二番目の引数が増加 `\c@section` するたびにリセットされます。二番目のカウンタはすでに定義されているものでな `\c@subsection` くてはいけません。

```

\c@subsubsection 1110 \newcounter{part}
1111 (*book | report)
\c@paragraph 1112 \newcounter{chapter}
\c@subparagraph 1113 \newcounter{section}[chapter]
1114 </book | report>
1115 <article>\newcounter{section}

```



```

1116 \newcounter{subsection}[section]
1117 \newcounter{subsubsection}[subsection]
1118 \newcounter{paragraph}[subsubsection]
1119 \newcounter{subparagraph}[paragraph]

```

`\thepart` `\theCTR` が実際に出力される形式の定義です。

```

\thechapter   \arabic{COUNTER}は、COUNTER の値を算用数字で出力します。
\thesection   \roman{COUNTER}は、COUNTER の値を小文字のローマ数字で出力します。
\thesubsection \Roman{COUNTER}は、COUNTER の値を大文字のローマ数字で出力します。
\thesubsubsection \alph{COUNTER}は、COUNTER の値を 1 = a, 2 = b のようにして出力します。
\theparagraph \Alph{COUNTER}は、COUNTER の値を 1 = A, 2 = B のようにして出力し
\thesubparagraph ます。

```

`\Kanji{COUNTER}`は、`COUNTER` の値を漢数字で出力します。

`\rensuji{obj}`は、`obj` を横に並べて出力します。したがって、横組のときには、何も影響しません。

```

1120 <*tate>
1121 \renewcommand{\thepart}{\rensuji{\@Roman\c@part}}
1122 <article>\renewcommand{\thesection}{\rensuji{\@arabic\c@section}}
1123 <*report | book>
1124 \renewcommand{\thechapter}{\rensuji{\@arabic\c@chapter}}
1125 \renewcommand{\thesection}{\thechapter.\rensuji{\@arabic\c@section}}
1126 </report | book>
1127 \renewcommand{\thesubsection}{\thesection.\rensuji{\@arabic\c@subsection}}
1128 \renewcommand{\thesubsubsection}{%
1129   \thesubsection.\rensuji{\@arabic\c@subsubsection}}
1130 \renewcommand{\theparagraph}{%
1131   \thesubsubsection.\rensuji{\@arabic\c@paragraph}}
1132 \renewcommand{\thesubparagraph}{%
1133   \theparagraph.\rensuji{\@arabic\c@subparagraph}}
1134 </tate>
1135 <*yoko>
1136 \renewcommand{\thepart}{\@Roman\c@part}
1137 <article>\renewcommand{\thesection}{\@arabic\c@section}
1138 <*report | book>
1139 \renewcommand{\thechapter}{\@arabic\c@chapter}
1140 \renewcommand{\thesection}{\thechapter.\@arabic\c@section}
1141 </report | book>
1142 \renewcommand{\thesubsection}{\thesection.\@arabic\c@subsection}
1143 \renewcommand{\thesubsubsection}{%
1144   \thesubsection.\@arabic\c@subsubsection}
1145 \renewcommand{\theparagraph}{%
1146   \thesubsubsection.\@arabic\c@paragraph}
1147 \renewcommand{\thesubparagraph}{%
1148   \theparagraph.\@arabic\c@subparagraph}
1149 </yoko>

```

`\@chapapp` `\@chapapp` の初期値は '`\prechaptername`' です。

`\@chappos` `\@chappos` の初期値は '`\postchaptername`' です。

`\appendix` コマンドは `\@chapapp` を '`\appendixname`' に、`\@chappos` を空に再定義します。

```
1150 (*report | book)
1151 \newcommand{\@chapapp}{\prechaptername}
1152 \newcommand{\@chappos}{\postchaptername}
1153 </report | book>
```

29.3.3 前付け、本文、後付け

`\frontmatter` 一冊の本は論理的に3つに分割されます。表題や目次や「はじめに」あるいは権利
`\mainmatter` などの前付け、そして本文、それから用語集や索引や奥付けなどの後付けです。

`\backmatter` **日本語 $T_E X$ 開発コミュニティによる補足**： \LaTeX の `classes.dtx` は、1996/05/26 (v1.3r) と 1998/05/05 (v1.3y) の計2回、`\frontmatter` と `\mainmatter` の定義を修正しています。一回目はこれらの命令を `openany` オプションに応じて切り替え、二回目はそれを元に戻しています。アスキーによる `jclasses.dtx` は、1997/01/15 に一回目の修正に追随しましたが、二回目の修正には追随していません。コミュニティ版では、一旦はアスキーによる仕様を維持しようと考えました (2016/11/22) が、以下の理由により二回目の修正にも追随することにしました (2017/03/05)。

アスキー版での `\frontmatter` と `\mainmatter` の改ページ挙動は

`openright` なら `\cleardoublepage`、`openany` なら `\clearpage` を実行

というものでした。しかし、`\frontmatter` 及び `\mainmatter` はノンブルを1にリセットしますから、改ページの結果が偶数ページ目になる場合⁴にノンブルが偶奇逆転してしまいました。このままでは `openany` の場合に両面印刷がうまくいかないため、新しいコミュニティ版では

必ず `\pltx@cleartooddpage` を実行

としました。これは両面印刷 (twoside) の場合は奇数ページに送り、片面印刷 (oneside) の場合は単に改ページとなります。(参考：[latex/2754](#))

```
1154 (*book)
1155 \newcommand{\frontmatter}{%
1156   \pltx@cleartooddpage
1157   \@mainmatterfalse\pagenumbering{roman}}
1158 \newcommand{\mainmatter}{%
1159   \pltx@cleartooddpage
1160   \@mainmattertrue\pagenumbering{arabic}}
1161 \newcommand{\backmatter}{%
1162   \if@openleft \cleardoublepage \else
1163   \if@openright \cleardoublepage \else \clearpage \fi \fi
1164   \@mainmatterfalse}
1165 </book>
```

⁴縦 tbook のデフォルト (`openright`) が該当するほか、横 jbook と縦 tbook の `openany` のときには成り行き次第で該当する可能性があります。

29.3.4 ボックスの組み立て

クラスファイル定義の、この部分では、`\@startsection` と `\secdef` の二つの内部マクロを使います。これらの構文を次に示します。

`\@startsection` マクロは6つの引数と1つのオプション引数 ‘*’ を取ります。

`\@startsection`
`\@startsection<name><level><indent><beforeskip><afterskip><style> optional *
 [<altheading>]<heading>`

それぞれの引数の意味は、次のとおりです。

`<name>` レベルコマンドの名前です (例:section)。

`<level>` 見出しの深さを示す数値です (chapter=1, section=2, ...)。 “`<level> <= カウンタ secnumdepth の値`” のとき、見出し番号が出力されます。

`<indent>` 見出しに対する、左マージンからのインデント量です。

`<beforeskip>` 見出しの上に置かれる空白の絶対値です。負の場合は、見出しに続くテキストのインデントを抑制します。

`<afterskip>` 正のとき、見出しの後の垂直方向のスペースとなります。負の場合は、見出しの後の水平方向のスペースとなります。

`<style>` 見出しのスタイルを設定するコマンドです。

`<*>` 見出し番号を付けないとき、対応するカウンタは増加します。

`<heading>` 新しい見出しの文字列です。

見出しコマンドは通常、`\@startsection` と6つの引数で定義されています。

`\secdef` マクロは、見出しコマンドを `\@startsection` を用いずに定義するときに使います。このマクロは、2つの引数を持ちます。

`\secdef`
`\secdef<unstarcmds><starcmds>`

`<unstarcmds>` 見出しコマンドの普通の形式で使われます。

`<starcmds>` *形式の見出しコマンドで使われます。

`\secdef` は次のようにして使うことができます。

```
\def\chapter {... \secdef \CMDA \CMDB }  
\def\CMDA    [#1]#2{...} % \chapter[...]{...} の定義  
\def\CMDB    #1{...}    % \chapter*{...} の定義
```

29.3.5 part レベル

`\part` このコマンドは、新しいパート（部）をはじめます。

article クラスの場合は、簡単です。

新しい段落を開始し、小さな空白を入れ、段落後のインデントを行い、`\secdef` で作成します。（アスキーによる元のドキュメントには「段落後のインデントをしないようにし」と書かれていましたが、実際のコードでは段落後のインデントを行っていました。そこで日本語 TeX 開発コミュニティは、ドキュメントをコードに合わせて「段落後のインデントを行い」へと修正しました。）

```
1166 (*article)
1167 \newcommand{\part}{%
1168   \if@noskipsec \leavevmode \fi
1169   \par\addvspace{4ex}%
1170   \@afterindenttrue
1171   \secdef\@part\@spart}
1172 \end{article}
```

report と book スタイルの場合は、少し複雑です。

まず、右ページからはじまるように改ページをします。そして、部扉のページスタイルを `empty` にします。2 段組の場合でも、1 段組で作成しますが、後ほど 2 段組に戻すために、`\@restonecol` スイッチを使います。

```
1173 (*report | book)
1174 \newcommand{\part}{%
1175   \if@openleft \cleardoublepage \else
1176   \if@openright \cleardoublepage \else \clearpage \fi \fi
1177   \thispagestyle{empty}%
1178   \if@twocolumn\onecolumn\@tempwatrue\else\@tempwafalse\fi
1179   \null\vfil
1180   \secdef\@part\@spart}
1181 \end{report | book}
```

`\@part` このマクロが実際に部レベルの見出しを作成します。このマクロも文書クラスによって定義が異なります。

article クラスの場合は、`secnumdepth` が -1 よりも大きいとき、見出し番号を付けます。このカウンタが -1 以下の場合には付けません。

```
1182 (*article)
1183 \def\@part[#1]#2{%
1184   \ifnum \c@secnumdepth >\m@ne
1185     \refstepcounter{part}%
1186     \addcontentsline{toc}{part}{%
1187       \prepartname\thepart\postpartname\hspace{1zw}#1}%
1188   \else
1189     \addcontentsline{toc}{part}{#1}%
1190   \fi
1191   \markboth{}{}%
1192   {\parindent\z@\raggedright
1193     \interlinepenalty\@M\normalfont
1194     \ifnum \c@secnumdepth >\m@ne
1195       \Large\bfseries\prepartname\thepart\postpartname
```

```

1196 \par\nobreak
1197 \fi
1198 \huge\bfseries#2\par}%
1199 \nobreak\vskip3ex\@afterheading}
1200 \</article>

```

report と book クラスの場合は、secnumdepth が -2 よりも大きいときに、見出し番号を付けます。-2 以下では付けません。

```

1201 \<report | book>
1202 \def\@part[#1]#2{%
1203 \ifnum \c@secnumdepth >-2\relax
1204 \refstepcounter{part}%
1205 \addcontentsline{toc}{part}{%
1206 \prepartname\thepart\postpartname\hspace{1em}#1}%
1207 \else
1208 \addcontentsline{toc}{part}{#1}%
1209 \fi
1210 \markboth{}{}%
1211 {\centering
1212 \interlinepenalty\@M\normalfont
1213 \ifnum \c@secnumdepth >-2\relax
1214 \huge\bfseries\prepartname\thepart\postpartname
1215 \par\vskip20\p@
1216 \fi
1217 \Huge\bfseries#2\par}%
1218 \@endpart}
1219 \</report | book>

```

\@spart このマクロは、番号を付けないときの体裁です。

```

1220 \<article>
1221 \def\@spart#1{%
1222 \parindent\z@\raggedright
1223 \interlinepenalty\@M\normalfont
1224 \huge\bfseries#1\par}%
1225 \nobreak\vskip3ex\@afterheading}
1226 \</article>

1227 \<report | book>
1228 \def\@spart#1{%
1229 \centering
1230 \interlinepenalty\@M\normalfont
1231 \Huge\bfseries#1\par}%
1232 \@endpart}
1233 \</report | book>

```

\@endpart \@part と \@spart の最後で実行されるマクロです。両面印刷モードのときは、白ページを追加します。二段組モードのときには、これ以降のページを二段組に戻します。2016 年 12 月から、openany のときに白ページを追加するのをやめました。このバグは L^AT_EX では classes.dtx v1.4b (2000/05/19) で修正されていました。(参考：latex/3155、texjporg/jsclasses#48)

```

1234 \<report | book>
1235 \def\@endpart{\vfil\newpage

```

```

1236 \if@twoside
1237 \if@openleft %% \if@openleft added (2017/02/15)
1238 \null\thispagestyle{empty}\newpage
1239 \else\if@openright %% \if@openright added (2016/12/18)
1240 \null\thispagestyle{empty}\newpage
1241 \fi\fi %% added (2016/12/18, 2017/02/15)
1242 \fi

```

二段組文書のとき、スイッチを二段組モードに戻す必要があります。

```

1243 \if@tempwa\twocolumn\fi}
1244 \report|book}

```

29.3.6 chapter レベル

chapter 章レベルは、必ずページの先頭から開始します。openright オプションが指定されている場合は、右ページからはじまるように `\cleardoublepage` を呼び出します。そうでなければ、`\clearpage` を呼び出します。なお、縦組の場合でも右ページからはじまるように、フォーマットファイルで `\clerdoublepage` が定義されています。

日本語 $T_E X$ 開発コミュニティによる補足：コミュニティ版の実装では、openright と openleft の場合に `\cleardoublepage` をクラスファイルの中で再々定義しています。27 を参照してください。

章見出しが出力されるページのスタイルは、jpl@in になります。jpl@in は、headnomble か footnomble のいずれかです。詳細は、第 28 節を参照してください。

また、\@topnum をゼロにして、章見出しの上にトップフロートが置かれないうにしています。

```

1245 \*report|book}
1246 \newcommand{\chapter}{%
1247 \if@openleft \cleardoublepage \else
1248 \if@openright \cleardoublepage \else \clearpage \fi \fi
1249 \thispagestyle{jpl@in}%
1250 \global\@topnum\z@
1251 \@afterindenttrue
1252 \secdef\@chapter\@schapter}

```

\@chapter このマクロは、章見出しに番号を付けるときに呼び出されます。secnumdepth が -1 よりも大きく、\@mainmatter が真 (book クラスの場合) のときに、番号を出力します。

日本語 $T_E X$ 開発コミュニティによる補足：本家 \LaTeX の classes では、二段組のときチャプタータイトルは一段組に戻されますが、アスキーによる jclasses では二段組のままにされています。したがって、チャプタータイトルより高い位置に右カラムの始点が来るといふ挙動になっていますが、コミュニティ版でもアスキー版の挙動を維持しています。

```

1253 \def\@chapter[#1]#2{%
1254 \ifnum \c@secnumdepth >\m@ne
1255 \book \if@mainmatter
1256 \refstepcounter{chapter}%
1257 \typeout{\@chapapp\space\thechapter\space\@chappos}%

```

```

1258 \addcontentsline{toc}{chapter}%
1259 {\protect\numberline{\@chapapp\thechapter\@chappos}{#1}}%
1260 \book \else\addcontentsline{toc}{chapter}{#1}\fi
1261 \else
1262 \addcontentsline{toc}{chapter}{#1}%
1263 \fi
1264 \chaptermark{#1}%
1265 \addtocontents{lof}{\protect\addvspace{10\p@}}%
1266 \addtocontents{lot}{\protect\addvspace{10\p@}}%
1267 \@makechapterhead{#2}\@afterheading}

```

`\@makechapterhead` このマクロが実際に章見出しを組み立てます。

```

1268 \def\@makechapterhead#1{\hbox}%
1269 \vskip2\Cvs
1270 {\parindent\z@
1271 \raggedright
1272 \normalfont\huge\bfseries
1273 \leavevmode
1274 \ifnum \c@secnumdepth >\m@ne
1275 \setlength\@tempdima{\linewidth}%
1276 \book \if@mainmatter
1277 \setbox\z@\hbox{\@chapapp\thechapter\@chappos\hskip1zw}%
1278 \addtolength\@tempdima{-\wd\z@}%
1279 \unhbox\z@\nobreak
1280 \book \fi
1281 \vtop{\hsize\@tempdima#1}%
1282 \else
1283 #1\relax
1284 \fi}\nobreak\vskip3\Cvs}

```

`\@schapter` このマクロは、章見出しに番号を付けないときに呼び出されます。

日本語 *TeX* 開発コミュニティによる補足：やはり二段組でチャプタータイトルより高い位置に右カラムの始点が来るといふ挙動を維持してあります。

```

1285 \def\@schapter#1{%
1286 \@makeschapterhead{#1}\@afterheading
1287 }

```

`\@makeschapterhead` 番号を付けない場合の形式です。

```

1288 \def\@makeschapterhead#1{\hbox}%
1289 \vskip2\Cvs
1290 {\parindent\z@
1291 \raggedright
1292 \normalfont\huge\bfseries
1293 \leavevmode
1294 \setlength\@tempdima{\linewidth}%
1295 \vtop{\hsize\@tempdima#1}\vskip3\Cvs}
1296 \report | book}

```

29.3.7 下位レベルの見出し

`\section` 見出しの前後に空白を付け、`\Large\bfseries` で出力をします。

```

1297 \newcommand{\section}{\@startsection{section}{1}{\z0}%
1298   {1.5\Cvs \@plus.5\Cvs \@minus.2\Cvs}%
1299   {.5\Cvs \@plus.3\Cvs}%
1300   {\normalfont\Large\bfseries}}

```

`\subsection` 見出しの前後に空白を付け、`\large\bfseries` で出力をします。

```

1301 \newcommand{\subsection}{\@startsection{subsection}{2}{\z0}%
1302   {1.5\Cvs \@plus.5\Cvs \@minus.2\Cvs}%
1303   {.5\Cvs \@plus.3\Cvs}%
1304   {\normalfont\large\bfseries}}

```

`\subsubsection` 見出しの前後に空白を付け、`\normalsize\bfseries` で出力をします。

```

1305 \newcommand{\subsubsection}{\@startsection{subsubsection}{3}{\z0}%
1306   {1.5\Cvs \@plus.5\Cvs \@minus.2\Cvs}%
1307   {.5\Cvs \@plus.3\Cvs}%
1308   {\normalfont\normalsize\bfseries}}

```

`\paragraph` 見出しの前に空白を付け、`\normalsize\bfseries` で出力をします。見出しの後ろで改行されません。

```

1309 \newcommand{\paragraph}{\@startsection{paragraph}{4}{\z0}%
1310   {3.25ex \@plus 1ex \@minus .2ex}%
1311   {-1em}%
1312   {\normalfont\normalsize\bfseries}}

```

`\subparagraph` 見出しの前に空白を付け、`\normalsize\bfseries` で出力をします。見出しの後ろで改行されません。

```

1313 \newcommand{\subparagraph}{\@startsection{subparagraph}{5}{\z0}%
1314   {3.25ex \@plus 1ex \@minus .2ex}%
1315   {-1em}%
1316   {\normalfont\normalsize\bfseries}}

```

29.3.8 付録

`\appendix article` クラスの場合、`\appendix` コマンドは次のことを行ないます。

- `section` と `subsection` カウンタをリセットする。
- `\thesection` を英小文字で出力するように再定義する。

```

1317 (*article)
1318 \newcommand{\appendix}{\par
1319   \setcounter{section}{0}%
1320   \setcounter{subsection}{0}%
1321   \tate \renewcommand{\thesection}{\rensujif{\@Alph{c@section}}}
1322   \yoko \renewcommand{\thesection}{\@Alph{c@section}}
1323 /article)

```

`report` と `book` クラスの場合、`\appendix` コマンドは次のことを行ないます。

- `chapter` と `section` カウンタをリセットする。

- \@chapapp を \appendixname に設定する。
- \@chappos を空にする。
- \thechapter を英小文字で出力するように再定義する。

```

1324 <*report | book>
1325 \newcommand{\appendix}{\par
1326   \setcounter{chapter}{0}%
1327   \setcounter{section}{0}%
1328   \renewcommand{\@chapapp}{\appendixname}%
1329   \renewcommand{\@chappos}{\space%
1330 <tate> \renewcommand{\thechapter}{\rensuji{\@Alph{c}{chapter}}}}
1331 <yoko> \renewcommand{\thechapter}{\@Alph{c}{chapter}}}
1332 </report | book>

```

29.4 リスト環境

ここではリスト環境について説明をしています。

リスト環境のデフォルトは次のように設定されます。

まず、\rightmargin, \listparindent, \itemindent をゼロにします。そして、K 番目のレベルのリストは \@listK で示されるマクロが呼び出されます。ここで ‘K’ は小文字のローマ数字で示されます。たとえば、3 番目のレベルのリストとして \@listiii が呼び出されます。 \@listK は \leftmargin を \leftmarginK に設定します。

\leftmargin 二段組モードのマージンは少しだけ小さく設定してあります。

```

\leftmargin1333 \if@twocolumn
\leftmarginii1334 \setlength\leftmargini {2em}
\leftmarginiii1335 \else
\leftmarginiiii1336 \setlength\leftmargini {2.5em}
\leftmarginiv1337 \fi

```

\leftmarginv 次の 3 つの値は、\labelsep とデフォルトラベル (‘(m)’, ‘vii.’, ‘M.’) の幅の合計より大きくしてあります。

```

1338 \setlength\leftmarginii {2.2em}
1339 \setlength\leftmarginiii {1.87em}
1340 \setlength\leftmarginiv {1.7em}
1341 \if@twocolumn
1342 \setlength\leftmarginv {.5em}
1343 \setlength\leftmarginvi{.5em}
1344 \else
1345 \setlength\leftmarginv {1em}
1346 \setlength\leftmarginvi{1em}
1347 \fi

```

\labelsep \labelsep はラベルとテキストの項目の間の距離です。 \labelwidth はラベルの幅 \labelwidth です。

```

1348 \setlength \labelsep {.5em}
1349 \setlength \labelwidth{\leftmargini}
1350 \addtolength\labelwidth{-\labelsep}

```

`\@beginparpenalty` これらのペナルティは、リストや段落環境の前後に挿入されます。

`\@endparpenalty`
`\@itempenalty` このペナルティは、リスト項目の間に挿入されます。

```
1351 \@beginparpenalty -\@lowpenalty
1352 \@endparpenalty -\@lowpenalty
1353 \@itempenalty -\@lowpenalty
1354 </article | report | book>
```

`\partopsep` リスト環境の前に空行がある場合、`\parskip` と `\topsep` に `\partopsep` が加えられた値の縦方向の空白が取られます。

```
1355 {10pt}\setlength\partopsep{2\p@ \@plus 1\p@ \@minus 1\p@}
1356 {11pt}\setlength\partopsep{3\p@ \@plus 1\p@ \@minus 1\p@}
1357 {12pt}\setlength\partopsep{3\p@ \@plus 2\p@ \@minus 2\p@}
```

`\@listi` `\@listi` は、`\leftmargin`、`\parsep`、`\topsep`、`\itemsep` などのトップレベルの定義をします。この定義は、フォントサイズコマンドによって変更されます（たとえば、`\small` の中では“小さい”リストパラメータになります）。

このため、`\normalsize` がすべてのパラメータを戻せるように、`\@listI` は `\@listi` のコピーを保存するように定義されています。

```
1358 (*10pt | 11pt | 12pt)
1359 \def\@listi{\leftmargin\leftmarginI
1360 (*10pt)
1361 \parsep 4\p@ \@plus2\p@ \@minus\p@
1362 \topsep 8\p@ \@plus2\p@ \@minus4\p@
1363 \itemsep4\p@ \@plus2\p@ \@minus\p@}
1364 </10pt>
1365 (*11pt)
1366 \parsep 4.5\p@ \@plus2\p@ \@minus\p@
1367 \topsep 9\p@ \@plus3\p@ \@minus5\p@
1368 \itemsep4.5\p@ \@plus2\p@ \@minus\p@}
1369 </11pt>
1370 (*12pt)
1371 \parsep 5\p@ \@plus2.5\p@ \@minus\p@
1372 \topsep 10\p@ \@plus4\p@ \@minus6\p@
1373 \itemsep5\p@ \@plus2.5\p@ \@minus\p@}
1374 </12pt>
1375 \let\@listI\@listi
```

ここで、パラメータを初期化しますが、厳密には必要ありません。

```
1376 \@listi
```

`\@listii` 下位レベルのリスト環境のパラメータの設定です。これらは保存用のバージョンを `\@listiii` 持たないことと、フォントサイズコマンドによって変更されないことに注意をして `\@listiv` ください。言い換えれば、このクラスは、本文サイズが `\normalsize` で現れるリストの入れ子についてだけ考えています。

```
\@listv1377 \def\@listii{\leftmargin\leftmarginii
1378 \labelwidth\leftmarginii \advance\labelwidth-\labelsep
1379 (*10pt)
1380 \topsep 4\p@ \@plus2\p@ \@minus\p@
1381 \parsep 2\p@ \@plus\p@ \@minus\p@
```

```

1382 </10pt>
1383 < *11pt>
1384   \topsep 4.5\p@ \@plus2\p@ \@minus\p@
1385   \parsep 2\p@   \@plus\p@   \@minus\p@
1386 </11pt>
1387 < *12pt>
1388   \topsep 5\p@   \@plus2.5\p@ \@minus\p@
1389   \parsep 2.5\p@ \@plus\p@   \@minus\p@
1390 </12pt>
1391   \itemsep\parsep}
1392 \def\@listiii{\leftmargin\leftmarginiii
1393   \labelwidth\leftmarginiii \advance\labelwidth-\labelsep
1394 <10pt>   \topsep 2\p@   \@plus\p@\@minus\p@
1395 <11pt>   \topsep 2\p@   \@plus\p@\@minus\p@
1396 <12pt>   \topsep 2.5\p@\@plus\p@\@minus\p@
1397   \parsep\z@
1398   \partopsep \p@ \@plus\z@ \@minus\p@
1399   \itemsep\topsep}
1400 \def\@listiv {\leftmargin\leftmarginiv
1401   \labelwidth\leftmarginiv
1402   \advance\labelwidth-\labelsep}
1403 \def\@listv  {\leftmargin\leftmarginv
1404   \labelwidth\leftmarginv
1405   \advance\labelwidth-\labelsep}
1406 \def\@listvi {\leftmargin\leftmarginvi
1407   \labelwidth\leftmarginvi
1408   \advance\labelwidth-\labelsep}
1409 </10pt | 11pt | 12pt>

```

29.4.1 enumerate 環境

enumerate 環境は、カウンタ *enumi*, *enumii*, *enumiii*, *enumiv* を使います。enum*N* は *N* 番目のレベルの番号を制御します。

`\theenumi` 出力する番号の書式を設定します。これらは、すでに `ltlists.dtx` で定義されています。
`\theenumii` ます。

```

\theenumiii 1410 <*article | report | book>
\theenumiv 1411 <*tate>
1412 \renewcommand{\theenumi}{\rensuji{\@arabic\c@enumi}}
1413 \renewcommand{\theenumii}{\rensuji{\@alph\c@enumii}}
1414 \renewcommand{\theenumiii}{\rensuji{\@roman\c@enumiii}}
1415 \renewcommand{\theenumiv}{\rensuji{\@Alph\c@enumiv}}
1416 </tate>
1417 <*yoko>
1418 \renewcommand{\theenumi}{\@arabic\c@enumi}
1419 \renewcommand{\theenumii}{\@alph\c@enumii}
1420 \renewcommand{\theenumiii}{\@roman\c@enumiii}
1421 \renewcommand{\theenumiv}{\@Alph\c@enumiv}
1422 </yoko>

```

`\labelenumi` enumerate 環境のそれぞれの項目のラベルは、`\labelenumi ... \labelenumiv` で
`\labelenumii` 生成されます。

`\labelenumiii`

`\labelenumiv` File h: jclasses.dtx

```

1423 <*tate>
1424 \newcommand{\labelenumi}{\theenumi}
1425 \newcommand{\labelenumii}{\theenumii}
1426 \newcommand{\labelenumiii}{\theenumiii}
1427 \newcommand{\labelenumiv}{\theenumiv}
1428 </tate>
1429 <*yoko>
1430 \newcommand{\labelenumi}{\theenumi.}
1431 \newcommand{\labelenumii}{(\theenumii)}
1432 \newcommand{\labelenumiii}{\theenumiii.}
1433 \newcommand{\labelenumiv}{\theenumiv.}
1434 </yoko>

```

`\p@enumii \ref` コマンドによって、`enumerate` 環境の N 番目のリスト項目が参照されるとき `\p@enumiii` の書式です。

```

\p@enumiv1435 \renewcommand{\p@enumii}{\theenumi}
1436 \renewcommand{\p@enumiii}{\theenumi(\theenumii)}
1437 \renewcommand{\p@enumiv}{\p@enumiii\theenumiii}

```

`enumerate (env.)` トップレベルで使われたときに、最初と最後に半行分のスペースを開けるように、変更します。この環境は、`ltlists.dtx` で定義されています。

```

1438 \renewenvironment{enumerate}
1439 {\ifnum \@enumdepth >\thr@@\toodeep\else
1440 \advance\@enumdepth\@ne
1441 \edef\@enumctr{enum\romannumeral\the\@enumdepth}%
1442 \expandafter \list \csname label\@enumctr\endcsname{%
1443 \iftdir
1444 \ifnum \@listdepth=\@ne \topsep.5\normalbaselineskip
1445 \else\topsep\z@\fi
1446 \parskip\z@ \itemsep\z@ \parsep\z@
1447 \labelwidth1zw \labelsep.3zw
1448 \ifnum \@enumdepth=\@ne \leftmargin1zw\relax
1449 \else\leftmargin\leftskip\fi
1450 \advance\leftmargin 1zw
1451 \fi
1452 \usecounter{\@enumctr}%
1453 \def\makelabel##1{\hss\llap{##1}}}%
1454 \fi}\endlist}

```

29.4.2 itemize 環境

`\labelitemi` `itemize` 環境のそれぞれの項目のラベルは、`\labelenumi ... \labelenumiv` で生成 `\labelitemii` されます。

```

\labelitemii1455 \newcommand{\labelitemi}{\labelitemfont \textbullet}
1456 \newcommand{\labelitemii}{%
\labelitemiv
1457 \iftdir
1458 {\labelitemfont \textcircled{~}}
1459 \else
1460 {\labelitemfont \bfseries\textendash}
1461 \fi
1462 }

```

```

1463 \newcommand{\labelitemiii}{\labelitemfont \textasteriskcentered}
1464 \newcommand{\labelitemiv}{\labelitemfont \textperiodcentered}
1465 \newcommand{\labelitemfont}{\normalfont}

```

`itemize (env.)` トップレベルで使われたときに、最初と最後に半行分のスペースを開けるように、変更します。この環境は、`ltlists.dtx` で定義されています。

```

1466 \renewenvironment{itemize}
1467 {\ifnum \@itemdepth >\thr@@\@toodeep\else
1468   \advance\@itemdepth\@ne
1469   \edef\@itemitem{\labelitem\romannumeral\the\@itemdepth}%
1470   \expandafter \list \csname \@itemitem\endcsname{%
1471     \iftdir
1472       \ifnum \@listdepth=\@ne \topsep.5\normalbaselineskip
1473       \else\topsep\z@\fi
1474       \parskip\z@ \itemsep\z@ \parsep\z@
1475       \labelwidth1zw \labelsep.3zw
1476       \ifnum \@itemdepth =\@ne \leftmargin1zw\relax
1477       \else\leftmargin\leftskip\fi
1478       \advance\leftmargin 1zw
1479     \fi
1480     \def\makelabel##1{\hss\llap{##1}}}%
1481   \fi}{\endlist}

```

29.4.3 description 環境

`description (env.)` `description` 環境を定義します。縦組時には、インデントが3字分だけ深くなります。

```

1482 \newenvironment{description}
1483 {\list{}{\labelwidth\z@ \itemindent-\leftmargin
1484   \iftdir
1485     \leftmargin\leftskip \advance\leftmargin3\Cwd
1486     \rightmargin\rightskip
1487     \labelsep=1zw \itemsep\z@
1488     \listparindent\z@ \topskip\z@ \parskip\z@ \partopsep\z@
1489   \fi
1490     \let\makelabel\descriptionlabel}}{\endlist}

```

`\descriptionlabel` ラベルの形式を変更する必要がある場合は、`\descriptionlabel` を再定義してください。

```

1491 \newcommand{\descriptionlabel}[1]{%
1492   \hspace\labelsep\normalfont\bfseries #1}

```

29.4.4 verse 環境

`verse (env.)` `verse` 環境は、リスト環境のパラメータを使って定義されています。改行をするには `\\` を用います。`\\` は `\@centercr` に `\let` されています。

```

1493 \newenvironment{verse}
1494 {\let\\ \@centercr
1495   \list{}{\itemsep\z@ \itemindent -1.5em%
1496     \listparindent\itemindent
1497     \rightmargin\leftmargin \advance\leftmargin 1.5em}%
1498   \item\relax}{\endlist}

```

29.4.5 quotation 環境

`quotation` (*env.*) `quotation` 環境もまた、`list` 環境のパラメータを使用して定義されています。この環境の各行は、`\textwidth` よりも小さく設定されています。この環境における、段落の最初の行はインデントされます。

```
1499 \newenvironment{quotation}
1500   {\list{}{\listparindent 1.5em%
1501             \itemindent\listparindent
1502             \rightmargin\leftmargin
1503             \parsep\z@ \@plus\p@}%
1504   \item\relax}{\endlist}
```

29.4.6 quote 環境

`quote` (*env.*) `quote` 環境は、段落がインデントされないことを除き、`quotation` 環境と同じです。

```
1505 \newenvironment{quote}
1506   {\list{}{\rightmargin\leftmargin}%
1507   \item\relax}{\endlist}
```

29.5 フロート

`lfloat.dtx` では、フロートオブジェクトを操作するためのツールしか定義していません。タイプが `TYPE` のフロートオブジェクトを扱うマクロを定義するには、次の変数が必要です。

`\fps@TYPE` タイプ `TYPE` のフロートを置くデフォルトの位置です。

`\ftype@TYPE` タイプ `TYPE` のフロートの番号です。各 `TYPE` には、一意な、2 の倍数の `TYPE` 番号を割り当てます。たとえば、図が番号 1 ならば、表は 2 です。次のタイプは 4 となります。

`\ext@TYPE` タイプ `TYPE` のフロートの目次を出力するファイルの拡張子です。たとえば、`\ext@figure` は `'lot'` です。

`\fnum@TYPE` キャプション用の図番号を生成するマクロです。たとえば、`\fnum@figure` は `'図 \thefigure'` を作ります。

29.5.1 figure 環境

ここでは、`figure` 環境を実装しています。

`\c@figure` 図番号です。

```
\thefigure508 <article>\newcounter{figure}
1509 <report | book>\newcounter{figure}[chapter]
1510 <*tate>
1511 <article>\renewcommand{\thefigure}{\rensuji{\@arabic\c@figure}}
```

```

1512 <*report | book>
1513 \renewcommand{\thefigure}{%
1514   \ifnum\c@chapter>\z@\thechapter{}\cdot\fi\rensuji{\@arabic\c@figure}}
1515 </report | book>
1516 </tate>
1517 <*yoko>
1518 <article>\renewcommand{\thefigure}{\@arabic\c@figure}
1519 <*report | book>
1520 \renewcommand{\thefigure}{%
1521   \ifnum\c@chapter>\z@\thechapter.\fi\@arabic\c@figure}
1522 </report | book>
1523 </yoko>

```

\fps@figure フロートオブジェクトタイプ “figure” のためのパラメータです。

```

\ftype@figure 1524 \def\fps@figure{tbp}
\ext@figure 1525 \def\ftype@figure{1}
1526 \def\ext@figure{lof}
\fnun@figure 1527 \def\fnun@figure{\figurename\thefigure}
1528 \def\fnun@figure{\figurename~\thefigure}

```

figure (env.) *形式は2段抜きのフロートとなります。

```

figure* (env.) 1529 \newenvironment{figure}
1530                 {\@float{figure}}
1531                 {\end@float}
1532 \newenvironment{figure*}
1533                 {\@dblfloat{figure}}
1534                 {\end@dblfloat}

```

29.5.2 table 環境

ここでは、table 環境を実装しています。

\c@table 表番号です。

```

\thetable 1535 <article>\newcounter{table}
1536 <report | book>\newcounter{table}[chapter]
1537 <*tate>
1538 <article>\renewcommand{\thetable}{\rensuji{\@arabic\c@table}}
1539 <*report | book>
1540 \renewcommand{\thetable}{%
1541   \ifnum\c@chapter>\z@\thechapter{}\cdot\fi\rensuji{\@arabic\c@table}}
1542 </report | book>
1543 </tate>
1544 <*yoko>
1545 <article>\renewcommand{\thetable}{\@arabic\c@table}
1546 <*report | book>
1547 \renewcommand{\thetable}{%
1548   \ifnum\c@chapter>\z@\thechapter.\fi\@arabic\c@table}
1549 </report | book>
1550 </yoko>

```

\fps@table フロートオブジェクトタイプ “table” のためのパラメータです。

```

\ftype@table 1551 \def\fps@table{tbp}
\ext@table
\fnun@table File h: jclasses.dtx

```

```

1552 \def\ftype@table{2}
1553 \def\ext@table{lot}
1554 \tate\def\fnun@table{\tablename\thetable}
1555 \yoko\def\fnun@table{\tablename~\thetable}

```

`table (env.)` *形式は2段抜きのフロートとなります。

```

table* (env.) 1556 \newenvironment{table}
1557             {\@float{table}}
1558             {\end@float}
1559 \newenvironment{table*}
1560             {\@dblfloat{table}}
1561             {\end@dblfloat}

```

29.6 キャプション

`\makecaption` `\caption` コマンドは、キャプションを組み立てるために `\mkcaption` を呼出ます。このコマンドは二つの引数を取ります。一つは、`<number>` で、フロートオブジェクトの番号です。もう一つは、`<text>` でキャプション文字列です。`<number>` には通常、‘図 3.2’ のような文字列が入っています。このマクロは、`\parbox` の中で呼び出されます。書体は `\normalsize` です。

`\abovecaptionskip` これらの長さはキャプションの前後に挿入されるスペースです。

```

\belowcaptionskip 1562 \newlength\abovecaptionskip
1563 \newlength\belowcaptionskip
1564 \setlength\abovecaptionskip{10\p@}
1565 \setlength\belowcaptionskip{0\p@}

```

キャプション内で複数の段落を作成することができるように、このマクロは `\long` で定義をします。

```

1566 \long\def\makecaption#1#2{%
1567   \vskip\abovecaptionskip
1568   \iftdir\sbox\@tempboxa{#1\hskip1zw#2}%
1569   \else\sbox\@tempboxa{#1: #2}%
1570   \fi
1571   \ifdim \wd\@tempboxa >\hsize
1572     \iftdir #1\hskip1zw#2\relax\par
1573     \else #1: #2\relax\par\fi
1574   \else
1575     \global \minipagefalse
1576     \hb@xt@\hsize{\hfil\box\@tempboxa\hfil}%
1577   \fi
1578   \vskip\belowcaptionskip}

```

29.7 コマンドパラメータの設定

29.7.1 array と tabular 環境

`\arraycolsep` array 環境のカラムは `2\arraycolsep` で分離されます。

```

1579 \setlength\arraycolsep{5\p@}

```


`\tabcolsep` tabular 環境のカラムは `2\tabcolsep` で分離されます。

```
1580 \setlength\tabcolsep{6\p@}
```

`\arrayrulewidth` array と tabular 環境内の罫線の幅です。

```
1581 \setlength\arrayrulewidth{.4\p@}
```

`\doublerulesep` array と tabular 環境内の罫線間を調整する空白です。

```
1582 \setlength\doublerulesep{2\p@}
```

29.7.2 tabbing 環境

`\tabbingsep` ‘`\`’ コマンドで置かれるスペースを制御します。

```
1583 \setlength\tabbingsep{\labelsep}
```

29.7.3 minipage 環境

`\@mpfootins` minipage にも脚注を付けることができます。`\skip\@mpfootins` は、通常の `\skip\footins` と同じような動作をします。

```
1584 \skip\@mpfootins = \skip\footins
```

29.7.4 framebox 環境

`\fboxsep` `\fboxsep` は、`\fbox` と `\framebox` での、テキストとボックスの間に入る空白です。

`\fboxrule` `\fboxrule` は `\fbox` と `\framebox` で作成される罫線の幅です。

```
1585 \setlength\fboxsep{3\p@}
```

```
1586 \setlength\fboxrule{.4\p@}
```

29.7.5 equation と eqnarray 環境

`\theequation` equation カウンタは、新しい章の開始でリセットされます。また、equation 番号には、章番号が付きます。

このコードは `\chapter` 定義の後、より正確には `chapter` カウンタの定義の後、でなくてははいけません。

```
1587 <article>\renewcommand{\theequation}{\@arabic\c@equation}
```

```
1588 <*report|book>
```

```
1589 \@addtoreset{equation}{chapter}
```

```
1590 \renewcommand{\theequation}{%
```

```
1591 \ifnum\c@chapter>\z@\thechapter.\fi \@arabic\c@equation}
```

```
1592 </report|book>
```

30 フォントコマンド

`disablejfam` オプションが指定されていない場合には、以下の設定がなされます。

まず、数式内に日本語を直接、記述するために数式記号用文字に “`JY1/mc/m/n`” を登録します。数式バージョンが `bold` の場合は、“`JY1/gt/m/n`” を用います。これらは、`\mathmc`、`\mathgt` として登録されます。また、日本語数式ファミリとして

`\symmincho`がこの段階で設定されます。`mathrmmc` オプションが指定されていた場合には、これに引き続き `\mathrm` と `\mathbf` を和欧文両対応にするための作業がなされます。この際、他のマクロとの衝突を避けるため `\AtBeginDocument` を用いて展開順序を遅らせる必要があります。

`disablejfam` オプションが指定されていた場合には、`\mathmc` と `\mathgt` に対してエラーを出すだけのダミーの定義を与える設定のみが行われます。

変更

p_{La}T_EX 2.09 compatibility mode では和文数式フォント fam が 2 重定義されていたので、その部分を変更しました。

```

1593 \if@enablejfam
1594   \if@compatibility\else
1595     \DeclareSymbolFont{mincho}{JY1}{mc}{m}{n}
1596     \DeclareSymbolFontAlphabet{\mathmc}{mincho}
1597     \SetSymbolFont{mincho}{bold}{JY1}{gt}{m}{n}
1598     \jfam\symmincho
1599     \DeclareMathAlphabet{\mathgt}{JY1}{gt}{m}{n}
1600   \fi
1601   \if@mathrmmc
1602     \AtBeginDocument{%
1603       \reDeclareMathAlphabet{\mathrm}{\mathrm}{\mathmc}
1604       \reDeclareMathAlphabet{\mathbf}{\mathbf}{\mathgt}
1605     }%
1606   \fi
1607 \else
1608   \DeclareRobustCommand{\mathmc}{%
1609     \@latex@error{Command \noexpand\mathmc invalid with\space
1610       'disablejfam' class option.}\@eha
1611   }
1612   \DeclareRobustCommand{\mathgt}{%
1613     \@latex@error{Command \noexpand\mathgt invalid with\space
1614       'disablejfam' class option.}\@eha
1615   }
1616 \fi

```

ここでは L^AT_EX 2.09 で一般的に使われていたコマンドを定義しています。これらのコマンドはテキストモードと数式モードの**どちらでも**動作します。これらは互換性のために提供をしますが、できるだけ `\text...` と `\math...` を使うようにしてください。

`\mc` これらのコマンドはフォントファミリーを変更します。互換モードの同名コマンドと `\gt` 異なり、すべてのコマンドがデフォルトフォントにリセットしてから、対応する属性を変更することに注意してください。

```

\sf 1617 \DeclareOldFontCommand{\mc}{\normalfont\mcfamily}{\mathmc}
1618 \DeclareOldFontCommand{\gt}{\normalfont\gtfamily}{\mathgt}
\tt 1619 \DeclareOldFontCommand{\rm}{\normalfont\rmfamily}{\mathrm}
1620 \DeclareOldFontCommand{\sf}{\normalfont\sffamily}{\mathsf}
1621 \DeclareOldFontCommand{\tt}{\normalfont\ttfamily}{\mathtt}

```

`\bf` このコマンドはボールド書体にします。ノーマル書体に変更するには、`\mdseries`

と指定をします。

```
1622 \DeclareOldFontCommand{\bf}{\normalfont\bfseries}{\mathbf}
```

`\it` これらのコマンドはフォントシェイプを切替えます。スラント体とスモールキャップの数式アルファベットはありませんので、数式モードでは何もしませんが、警告メッセージを出力します。`\upshape` コマンドで通常のシェイプにすることができます。

```
1623 \DeclareOldFontCommand{\it}{\normalfont\itshape}{\mathit}
```

```
1624 \DeclareOldFontCommand{\sl}{\normalfont\slshape}{\@nomath\sl}
```

```
1625 \DeclareOldFontCommand{\sc}{\normalfont\scshape}{\@nomath\sc}
```

`\cal` これらのコマンドは数式モードでだけ使うことができます。数式モード以外では何もしません。現在の NFSS は、これらのコマンドが警告を生成するように定義していますので、‘手ずから’定義する必要があります。

```
1626 \DeclareRobustCommand*\cal{\@fontswitch\relax\mathcal}
```

```
1627 \DeclareRobustCommand*\mit{\@fontswitch\relax\mathnormal}
```

31 相互参照

31.1 目次

`\section` コマンドは、`.toc` ファイルに、次のような行を出力します。

```
\contentsline{section}{<title>}{<page>}
```

`<title>` には項目が、`<page>` にはページ番号が入ります。`\section` に見出し番号が付く場合は、`<title>` は、`\numberline{<num>}{<heading>}` となります。`<num>` は `\thesection` コマンドで生成された見出し番号です。`<heading>` は見出し文字列です。この他の見出しコマンドも同様です。

figure 環境での `\caption` コマンドは、`.lof` ファイルに、次のような行を出力します。

```
\contentsline{figure}{\numberline{<num>}{<caption>}}{<page>}
```

`<num>` は、`\thefigure` コマンドで生成された図番号です。`<caption>` は、キャプション文字列です。table 環境も同様です。

`\contentsline{<name>}` コマンドは、`\l@<name>` に展開されます。したがって、目次の体裁を記述するには、`\l@chapter`、`\l@section` などを定義します。図目次のためには `\l@figure` です。これらの多くのコマンドは `\@dottedtocline` コマンドで定義されています。このコマンドは次のような書式となっています。

```
\@dottedtocline{<level>}{<indent>}{<numwidth>}{<title>}{<page>}
```

`<level>` “`<level> <= tocdepth`” のときにだけ、生成されます。`\chapter` はレベル 0、`\section` はレベル 1、... です。

`<indent>` 一番外側からの左マージンです。

$\langle numwidth \rangle$ 見出し番号 (`\numberline` コマンドの $\langle num \rangle$) が入るボックスの幅です。

`\c@tocdepth` *tocdepth* は、目次ページに出力をする見出しレベルです。

```
1628 <article>\setcounter{tocdepth}{3}
1629 <!article>\setcounter{tocdepth}{2}
```

また、目次を生成するために次のパラメータも使います。

`\@pnumwidth` ページ番号の入るボックスの幅です。

```
1630 \newcommand{\@pnumwidth}{1.55em}
```

`\@tocrmarg` 複数行にわたる場合の右マージンです。

```
1631 \newcommand{\@tocrmarg}{2.55em}
```

`\@dotsep` ドットの間隔 (μ 単位) です。2 や 1.7 のように指定をします。

```
1632 \newcommand{\@dotsep}{4.5}
```

`\toclineskip` この長さ変数は、目次項目の間に入るスペースの長さです。デフォルトはゼロとなっています。縦組のとき、スペースを少し広げます。

```
1633 \newdimen\toclineskip
1634 <yoko>\setlength\toclineskip{z@}
1635 <tate>\setlength\toclineskip{2p@}
```

`\numberline` `\numberline` マクロの定義を示します。オリジナルの定義では、ボックスの幅を `\@lnumwidth` `\@tempdima` にしていますが、この変数はいろいろな箇所で使われますので、期待した値が入らない場合があります。

フォント選択コマンドの後、あるいは `\numberline` マクロの中でフォントを切替えてもよいのですが、一時変数を意識したくないので、見出し番号の入るボックスを `\@lnumwidth` 変数を用いて組み立てるように `\numberline` マクロを再定義します。

```
1636 \newdimen\@lnumwidth
1637 \def\numberline#1{\hb@xt@\@lnumwidth{#1\hfil}}
```

`\@dottedtocline` 目次の各行間に `\toclineskip` を入れるように変更します。このマクロは `ltsect.dtx` で定義されています。

```
1638 \def\@dottedtocline#1#2#3#4#5{%
1639   \ifnum #1>\c@tocdepth \else
1640     \vskip\toclineskip \@plus.2\p@
1641     {\leftskip #2\relax \rightskip \@tocrmarg \parfillskip -\rightskip
1642     \parindent #2\relax\@afterindenttrue
1643     \interlinepenalty\@M
1644     \leavevmode
1645     \@lnumwidth #3\relax
1646     \advance\leftskip \@lnumwidth \null\nobreak\hskip -\leftskip
1647     {#4}\nobreak
1648     \leaders\hbox{$\m@th \mkern \@dotsep mu.\mkern \@dotsep mu$}%
1649     \hfill\nobreak
```

```

1650 <yoko>      \hb@xt@{\pnumwidth{\hss\normalfont \normalcolor #5}}%
1651 <tate>      \hb@xt@{\pnumwidth{\hss\normalfont \normalcolor\rensuji{#5}}}%
1652      \par}%
1653 \fi}

```

`\addcontentsline` 以前のバージョンでは縦組の場合にページ番号を `\rensuji` で囲むように再定義していました。これでは `hyperref` パッケージと干渉してしまうので、`\rensuji` は `\l@chapter` など実際の出力命令のほうに押し込めることにし、`\addcontentsline` の再定義は削除します。

31.1.1 本文目次

`\tableofcontents` 目次を生成します。

```

1654 \newcommand{\tableofcontents}{%
1655   (*report | book)
1656   \if@twocolumn\@restonecoltrue\onecolumn
1657   \else\@restonecolfalse\fi
1658   /report | book)
1659   <article> \section*{\contentsname
1660   !article> \chapter*{\contentsname

```

`\tableofcontents` では、`\@mkboth` は heading の中に入れてあります。ほかの命令 (`\listoffigures` など) については、`\@mkboth` は heading の外に出してあります。これは L^AT_EX の `classes.dtx` に合わせています。

```

1661   \@mkboth{\contentsname}{\contentsname}%
1662   }\@starttoc{toc}%
1663   <report | book> \if@restonecol\twocolumn\fi
1664 }

```

`\l@part` part レベルの目次です。

```

1665 \newcommand*{\l@part}[2]{%
1666   \ifnum \c@tocdepth >-2\relax
1667   <article> \addpenalty{\@secpenalty}%
1668   !article> \addpenalty{-\@highpenalty}%
1669   \addvspace{2.25em \@plus\p@}%
1670   \begingroup
1671   \parindent\z@\rightskip\@pnumwidth
1672   \parfillskip-\@pnumwidth
1673   {\leavevmode\large\bfseries
1674   \setlength{\lnumwidth}{4zw}%
1675   #1\hfil\nobreak
1676   <yoko>      \hb@xt@{\pnumwidth{\hss#2}}\par
1677   <tate>      \hb@xt@{\pnumwidth{\hss\rensuji{#2}}}\par
1678   \nobreak
1679   <article> \if@compatibility
1680   \global\@nobreaktrue
1681   \everypar{\global\@nobreakfalse\everypar{}}%
1682   <article> \fi
1683   \endgroup
1684   \fi}

```

`\l@chapter` chapter レベルの目次です。

```
1685 <*report | book>
1686 \newcommand*{\l@chapter}[2]{%
1687   \ifnum \c@tocdepth >\m@ne
1688     \addpenalty{-\@highpenalty}%
1689     \advvspace{1.0em \@plus\p@}%
1690     \begingroup
1691       \parindent\z@ \rightskip\@pnumwidth \parfillskip-\rightskip
1692       \leavevmode\bfseries
1693       \setlength\@lnumwidth{4zw}%
1694       \advance\leftskip\@lnumwidth \hskip-\leftskip
1695 (yoko)       #1\nobreak\hfil\nobreak\hb@xt@\@pnumwidth{\hss#2}\par
1696 (tate)       #1\nobreak\hfil\nobreak\hb@xt@\@pnumwidth{\hss\rensuji{#2}}\par
1697       \penalty\@highpenalty
1698     \endgroup
1699   \fi}
1700 </report | book>
```

`\l@section` section レベルの目次です。

```
1701 <*article>
1702 \newcommand*{\l@section}[2]{%
1703   \ifnum \c@tocdepth >\z@
1704     \addpenalty{\@secpenalty}%
1705     \advvspace{1.0em \@plus\p@}%
1706     \begingroup
1707       \parindent\z@ \rightskip\@pnumwidth \parfillskip-\rightskip
1708       \leavevmode\bfseries
1709       \setlength\@lnumwidth{1.5em}%
1710       \advance\leftskip\@lnumwidth \hskip-\leftskip
1711 (yoko)       #1\nobreak\hfil\nobreak\hb@xt@\@pnumwidth{\hss#2}\par
1712 (tate)       #1\nobreak\hfil\nobreak\hb@xt@\@pnumwidth{\hss\rensuji{#2}}\par
1713     \endgroup
1714   \fi}
1715 </article>

1716 <*report | book>
1717 (tate) \newcommand*{\l@section}{\@dottedtocline{1}{1zw}{4zw}}
1718 (yoko) \newcommand*{\l@section}{\@dottedtocline{1}{1.5em}{2.3em}}
1719 </report | book>
```

`\l@subsection` 下位レベルの目次項目の体裁です。

```
\l@subsubsection1720 <*tate>
\l@paragraph1721 <*article>
\l@subparagraph1722 \newcommand*{\l@subsection} {\@dottedtocline{2}{1zw}{4zw}}
1723 \newcommand*{\l@subsubsection}{\@dottedtocline{3}{2zw}{6zw}}
1724 \newcommand*{\l@paragraph} {\@dottedtocline{4}{3zw}{8zw}}
1725 \newcommand*{\l@subparagraph} {\@dottedtocline{5}{4zw}{9zw}}
1726 </article>
1727 <*report | book>
1728 \newcommand*{\l@subsection} {\@dottedtocline{2}{2zw}{6zw}}
1729 \newcommand*{\l@subsubsection}{\@dottedtocline{3}{3zw}{8zw}}
1730 \newcommand*{\l@paragraph} {\@dottedtocline{4}{4zw}{9zw}}
1731 \newcommand*{\l@subparagraph} {\@dottedtocline{5}{5zw}{10zw}}
```

```

1732 </report | book>
1733 </tate>
1734 <*yoko>
1735 <*article>
1736 \newcommand*{\l@section}    {\@dottedtocline{2}{1.5em}{2.3em}}
1737 \newcommand*{\l@subsubsection}{\@dottedtocline{3}{3.8em}{3.2em}}
1738 \newcommand*{\l@paragraph}   {\@dottedtocline{4}{7.0em}{4.1em}}
1739 \newcommand*{\l@subparagraph}{\@dottedtocline{5}{10em}{5em}}
1740 </article>
1741 <*report | book>
1742 \newcommand*{\l@section}    {\@dottedtocline{2}{3.8em}{3.2em}}
1743 \newcommand*{\l@subsubsection}{\@dottedtocline{3}{7.0em}{4.1em}}
1744 \newcommand*{\l@paragraph}   {\@dottedtocline{4}{10em}{5em}}
1745 \newcommand*{\l@subparagraph}{\@dottedtocline{5}{12em}{6em}}
1746 </report | book>
1747 </yoko>

```

31.1.2 図目次と表目次

`\listoffigures` 図の一覧を作成します。

```

1748 \newcommand{\listoffigures}{%
1749 <*report | book>
1750   \if@twocolumn\@restonecoltrue\onecolumn
1751   \else\@restonecolfalse\fi
1752   \chapter*{\listfigurename}%
1753 </report | book>
1754 <article>   \section*{\listfigurename}%
1755   \mkboth{\listfigurename}{\listfigurename}%
1756   \starttoc{lof}%
1757 <report | book> \if@restonecol\twocolumn\fi
1758 }

```

`\l@figure` 図目次の体裁です。

```

1759 <tate> \newcommand*{\l@figure}{\@dottedtocline{1}{1zw}{4zw}}
1760 <yoko> \newcommand*{\l@figure}{\@dottedtocline{1}{1.5em}{2.3em}}

```

`\listoftables` 表の一覧を作成します。

```

1761 \newcommand{\listoftables}{%
1762 <*report | book>
1763   \if@twocolumn\@restonecoltrue\onecolumn
1764   \else\@restonecolfalse\fi
1765   \chapter*{\listtablename}%
1766 </report | book>
1767 <article>   \section*{\listtablename}%
1768   \mkboth{\listtablename}{\listtablename}%
1769   \starttoc{lot}%
1770 <report | book> \if@restonecol\twocolumn\fi
1771 }

```

`\l@table` 表目次の体裁は、図目次と同じにします。

```

1772 \let\l@table\l@figure

```

31.2 参考文献

`\bibindent` オープンスタイルの参考文献で使うインデント幅です。

```
1773 \newdimen\bibindent
1774 \setlength\bibindent{1.5em}
```

`\newblock` `\newblock` のデフォルト定義は、小さなスペースを生成します。

```
1775 \newcommand{\newblock}{\hskip .11em\@plus.33em\@minus.07em}
```

`thebibliography` (*env.*) 参考文献や関連図書のリストを作成します。

```
1776 \newenvironment{thebibliography}[1]
1777 <article>{\section*{\refname}\@mkboth{\refname}{\refname}}%
1778 <report | book>{\chapter*{\bibname}\@mkboth{\bibname}{\bibname}}%
1779 \list{\@biblabel{\@arabic\c@enumiv}}%
1780 \settolwidth\labelwidth{\@biblabel{#1}}%
1781 \leftmargin\labelwidth
1782 \advance\leftmargin\labelsep
1783 \@openbib@code
1784 \usecounter{enumiv}%
1785 \let\p@enumiv\@empty
1786 \renewcommand\theenumiv{\@arabic\c@enumiv}}%
1787 \sloppy

1788 \clubpenalty4000
1789 \@clubpenalty\clubpenalty
1790 \widowpenalty4000%
1791 \sfcode'\.\@m}
1792 {\def\@noitemerr
1793 {\@latex@warning{Empty 'thebibliography' environment}}}%
1794 \endlist}
```

`\@openbib@code` `\@openbib@code` のデフォルト定義は何もしません。この定義は、`openbib` オプションによって変更されます。

```
1795 \let\@openbib@code\@empty
```

`\@biblabel` The label for a `\bibitem[...]` command is produced by this macro. The default from `latex.dtx` is used.

```
1796 % \renewcommand*{\@biblabel}[1]{[#1]\hfill}
```

`\@cite` The output of the `\cite` command is produced by this macro. The default from `ltxbib1.dtx` is used.

```
1797 % \renewcommand*{\@cite}[1]{[#1]}
```

31.3 索引

`theindex` (*env.*) 2 段組の索引を作成します。索引の先頭のページのスタイルは `jpl@in` とします。したがって、`headings` と `bothstyle` に適した位置に出力されます。

```
1798 \newenvironment{theindex}
1799 {\if@twocolumn\@restonecolfalse\else\@restonecoltrue\fi
1800 <article> \twocolumn[\section*{\indexname}]}%
```



```

1801 <report | book> \twocolumn[@makeschapterhead{indexname}]%
1802 \mkboth{indexname}{indexname}%
1803 \thispagestyle{jpl@in}\parindent\z@

```

パラメータ `\columnseprule` と `\columnsep` の変更は、`\twocolumn` が実行された後でなければなりません。そうしないと、索引の前のページにも影響してしまうためです。

```

1804 \parskip\z@ \@plus .3\p@\relax
1805 \columnseprule\z@ \columnsep 35\p@
1806 \let\item\@idxitem}
1807 {\if@restonecol\onecolumn\else\clearpage\fi}

```

`\@idxitem` 索引項目の字下げ幅です。`\@idxitem` は `\item` の項目の字下げ幅です。

```

\subitem1808 \newcommand{\@idxitem}{\par\hangindent 40\p@}
1809 \newcommand{\subitem}{\@idxitem \hspace*{20\p@}}
\subsubitem1810 \newcommand{\subsubitem}{\@idxitem \hspace*{30\p@}}

```

`\indexspace` 索引の“文字”見出しの前に入るスペースです。

```

1811 \newcommand{\indexspace}{\par \vskip 10\p@ \@plus5\p@ \@minus3\p@\relax}

```

31.4 脚注

`\footnoterule` 本文と脚注の間に引かれる罫線です。

```

1812 \renewcommand{\footnoterule}{%
1813 \kern-3\p@
1814 \hrule\@width.4\columnwidth
1815 \kern2.6\p@}

```

`\c@footnote` report と book クラスでは、chapter レベルでリセットされます。

```

1816 {\article}\@addtoreset{footnote}{chapter}

```

`\@makefnmark` このマクロにしたがって脚注が組まれます。

`\@makefnmark` は脚注記号を組み立てるマクロです。

```

1817 <*tate>
1818 \newcommand\@makefnmark[1]{\parindent 1zw
1819 \noindent\hb@xt@ 2zw{\hss\@makefnmark}#1}
1820 </tate>
1821 <*yoko>
1822 \newcommand\@makefnmark[1]{\parindent 1em
1823 \noindent\hb@xt@ 1.8em{\hss\@makefnmark}#1}
1824 </yoko>

```

32 今日の日付

組版時における現在の日付を出力します。

`\if 西暦 \today` コマンドの‘年’を、西暦か和暦のどちらで出力するかを指定するコマンド `\ 西暦` です。2018 年 7 月以降の日本語 $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ 開発コミュニティ版 (v1.8) では、デフォルト `\ 和暦` を和暦ではなく西暦に設定しています。

```

1825 \newif\if 西暦 \西暦 true
1826 \def\西暦{\西暦 true}
1827 \def\和暦{\西暦 false}

```

\heisei \today コマンドを \rightmark で指定したとき、\rightmark を出力する部分で和暦のための計算ができないので、クラスファイルを読み込む時点で計算しておきます。

```

1828 \newcount\heisei \heisei\year \advance\heisei-1988\relax

```

\today 縦組の場合は、漢数字で出力します p_{LA}T_EX 2018-12-01 以前では縦数式ディレクション時でも漢数字で出力していましたが、p_{LA}T_EX 2019-04-06 以降からはそうしなくなりました。

```

1829 \def\pltx@today@year@#1{%
1830   \ifnum\numexpr\year-#1=1 元 \else
1831     \ifnum1=\iftdir\ifmdir0\else1\fi\else0\fi
1832     \kansuji\number\numexpr\year-#1\relax
1833   \else
1834     \number\numexpr\year-#1\relax\nobreak
1835   \fi
1836   \fi 年
1837 }
1838 \def\pltx@today@year{%
1839   \ifnum\numexpr\year*10000+\month*100+\day<19890108
1840     昭和 \pltx@today@year@{1925}%
1841   \else\ifnum\numexpr\year*10000+\month*100+\day<20190501
1842     平成 \pltx@today@year@{1988}%
1843   \else
1844     令和 \pltx@today@year@{2018}%
1845   \fi\fi}
1846 \def\today{%
1847   \if 西暦
1848     \ifnum1=\iftdir\ifmdir0\else1\fi\else0\fi\kansuji\number\year
1849     \else\number\year\nobreak\fi 年
1850   \else
1851     \pltx@today@year
1852   \fi
1853   \ifnum1=\iftdir\ifmdir0\else1\fi\else0\fi
1854   \kansuji\number\month 月
1855   \kansuji\number\day 日
1856   \else
1857     \number\month\nobreak 月
1858     \number\day\nobreak 日
1859   \fi}}

```

33 初期設定

```

\prepartname
\postpartname
\prechaptername
\postchaptername
1860 \newcommand{\prepartname}{第}
1861 \newcommand{\postpartname}{部}
1862 \newcommand{\prechaptername}{第}
1863 \newcommand{\postchaptername}{章}

```

```

\contentsname
\listfigurename1864 \newcommand{\contentsname}{目 次}
\listtablename1865 \newcommand{\listfigurename}{図 目 次}
1866 \newcommand{\listtablename}{表 目 次}

\refname
\bibname1867 <article> \newcommand{\refname}{参考文献}
\indexname1868 <report | book> \newcommand{\bibname}{関連図書}
1869 \newcommand{\indexname}{索 引}

\figurename
\tablename1870 \newcommand{\figurename}{図}
1871 \newcommand{\tablename}{表}

\appendixname
\abstractname1872 \newcommand{\appendixname}{付 録}
1873 <article | report> \newcommand{\abstractname}{概 要}

1874 <book> \pagestyle{headings}
1875 !<book> \pagestyle{plain}
1876 \pagenumbering{arabic}
1877 \raggedbottom
1878 \if@twocolumn
1879 \twocolumn
1880 \sloppy
1881 \else
1882 \onecolumn
1883 \fi

```

\@mparswitch は傍注を左右（縦組では上下）どちらのマージンに出力するかの指定です。偽の場合、傍注は一方の側にしか出力されません。このスイッチを真とすると、とくに縦組の場合、奇数ページでは本文の上に、偶数ページでは本文の下に傍注が出力されますので、おかしいことになります。

また、縦組のときには、傍注を本文の下に出すようにしています。reversemarginpar とすると本文の上側に出力されます。ただし、二段組の場合は、つねに隣接するテキスト側のマージンに出力されます。

```

1884 <*tate>
1885 \normalmarginpar
1886 \@mparswitchfalse
1887 </tate>
1888 <*yoko>
1889 \if@twoside
1890 \@mparswitchtrue
1891 \else
1892 \@mparswitchfalse
1893 \fi
1894 </yoko>
1895 </article | report | book>

```

File i

jltxdoc.dtx

jltxdoc クラスは、ltxdoc をテンプレートにして、日本語用の修正を加えています。

```
1 (*class)
2 \DeclareOption*{\PassOptionsToClass{\CurrentOption}{ltxdoc}}
3 \ProcessOptions
4 \LoadClass{ltxdoc}
```

`\normalsize` ltxdoc からロードされる article クラスでの行間などの設定値で、日本語の文章
`\small` を組版すると、行間が狭いように思われるので、多少広くするように再設定します。
`\parindent` また、段落先頭での字下げ量を全角一文字分とします。

```
5 \renewcommand{\normalsize}{%
6   \setfontsize\normalsize\@xpt{15}%
7   \abovedisplayskip 10\p@ \@plus2\p@ \@minus5\p@
8   \abovedisplayshortskip \z@ \@plus3\p@
9   \belowdisplayshortskip 6\p@ \@plus3\p@ \@minus3\p@
10  \belowdisplayskip \abovedisplayskip
11  \let\@listi\@listI}
12 \renewcommand{\small}{%
13   \setfontsize\small\@ixpt{11}%
14   \abovedisplayskip 8.5\p@ \@plus3\p@ \@minus4\p@
15   \abovedisplayshortskip \z@ \@plus2\p@
16   \belowdisplayshortskip 4\p@ \@plus2\p@ \@minus2\p@
17   \def\@listi{\leftmargin\leftmarginI
18     \topsep 4\p@ \@plus2\p@ \@minus2\p@
19     \parsep 2\p@ \@plus\p@ \@minus\p@
20     \itemsep \parsep}%
21   \belowdisplayskip \abovedisplayskip}
22 \normalsize
23 \setlength\parindent{1zw}
```

`\file` `\file` マクロは、ファイル名を示すのに用います。

```
24 \providecommand*{\file}[1]{\texttt{#1}}
```

`\pstyle` `\pstyle` マクロは、ページスタイル名を示すのに用います。

```
25 \providecommand*{\pstyle}[1]{\textsl{#1}}
```

`\Lcount` `\Lcount` マクロは、カウンタ名を示すのに用います。

```
26 \providecommand*{\Lcount}[1]{\textsl{\small#1}}
```

`\Lopt` `\Lopt` マクロは、クラスオプションやパッケージオプションを示すのに用います。

```
27 \providecommand*{\Lopt}[1]{\textsf{#1}}
```

`\dst` `\dst` マクロは、“DOCSTRIP” を出力する。

```
28 \providecommand\dst{\normalfont\scshape docstrip}}
```

`\NFSS` `\NFSS` マクロは、“NFSS” を出力します。

```
29 \providecommand\NFSS{\textsf{NFSS}}
```

`\c@clinenno` `\mlineplus` マクロは、その時点でのマクロコードの行番号に、引数に指定された `\mlineplus` 行数だけを加えた数値を出力します。たとえば `\mlineplus{3}` とすれば、直前のマクロコードの行番号 (29) に 3 を加えた数、“32” が出力されます。

```
30 \newcounter{@clinenno}
31 \def\mlineplus#1{\setcounter{@clinenno}{\arabic{CodelineNo}}}%
32 \addtocounter{@clinenno}{#1}\arabic{@clinenno}}
```

`tsample (env.)` `tsample` 環境は、環境内に指定された内容を罫線で囲って出力をします。第一引数は、出力するボックスの高さです。`plext.dtx` の中で使用しています。このマクロ内では縦組になることに注意してください。

```
33 \def\tsample#1{%
34 \hbox to\linewidth\bgroup\vrule width.1pt\hss
35 \vbox\bgroup\hrule height.1pt
36 \vskip.5\baselineskip
37 \vbox to\linewidth\bgroup\tate\hsize=#1\relax\vss}
38 \def\endtsample{%
39 \vss\egroup
40 \vskip.5\baselineskip
41 \hrule height.1pt\egroup
42 \hss\vrule width.1pt\egroup}
```

`\DisableCrossrefs` `jclasses.dtx` を処理するとき、`\if` 西暦の部分でエラーになるため、一時的に `\EnableCrossrefs` クロスリファレンスの機能をオフにします。しかし、デフォルトの定義では完全に制御できないので、ここで再定義をします。

```
43 \def\DisableCrossrefs{\@bsphack\scan@allowedfalse\@esphack}
44 \def\EnableCrossrefs{\@bsphack\scan@allowedtrue
45 \def\DisableCrossrefs{\@bsphack\scan@allowedfalse\@esphack}\@esphack}
```

`\verb` pL^AT_EX では、`\verb` コマンドを修正して直前に `\xkanjiskip` が入るようにしています。しかし、`ltxdoc.cls` が読み込む `doc.sty` が上書きしてしまいますので、これを再々定義します。`doc.sty` での定義は

```
\def\verb{\relax\ifmmode\hbox\else\leavevmode\null\fi
\bggroup \let\do\do@noligs \verbatim@nolig@list
\ttfamily \verb@eol@error \let\do\@makeother \dospecials
\@ifstar{\@sverb}{\@vobeyspaces \frenchspacing \@sverb}}
```

となっていますので、`plcore.dtx` と同様に `\null` を外して `\vadjust{}` を入れます。

```
46 \def\verb{\relax\ifmmode\hbox\else\leavevmode\vadjust{}\fi
47 \bggroup \let\do\do@noligs \verbatim@nolig@list
48 \ttfamily \verb@eol@error \let\do\@makeother \dospecials
49 \@ifstar{\@sverb}{\@vobeyspaces \frenchspacing \@sverb}}
```

`\xspcode` コマンド名の `\` と 16 進数を示すための `"` の前にもスペースが入るよう、これらの `\xspcode` の値を変更します。

```
50 \xspcode"5C=3 %% \
51 \xspcode"22=3 %% "
52 \</class>
```