

# pL<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> 新ドキュメントクラス

奥村晴彦, 日本語 T<sub>E</sub>X 開発コミュニティ

2020/10/05

## 1 はじめに

これは L<sup>A</sup>T<sub>E</sub>X3 Project の `classes.dtx` と株式会社アスキーの `jclasses.dtx` に基づいてもともと奥村晴彦により作成されたものです。現在は日本語 T<sub>E</sub>X 開発コミュニティにより GitHub で管理されています。

<https://github.com/texjporg/jsclasses>

[2002-12-19] いろいろなものに収録していただく際にライセンスを明確にする必要が生じてきました。アスキーのものが最近では modified BSD ライセンスになっていますので、私のものもそれに準じて modified BSD とすることにします。

[2016-07-13] 日本語 T<sub>E</sub>X 開発コミュニティによる管理に移行しました。

[2009-02-22] 田中琢爾氏による upL<sup>A</sup>T<sub>E</sub>X 対応パッチを取り込みました。

ここでは次のドキュメントクラス（スタイルファイル）を作ります。

[2017-02-13] forum:2121 の議論を機に、jsreport クラスを新設しました。従来の jsbook の report オプションと比べると、abstract 環境の使い方および挙動がアスキーの jreport に近づきました。

<code>&lt;article&gt;</code>	<code>jsarticle.cls</code>	論文・レポート用
<code>&lt;book&gt;</code>	<code>jsbook.cls</code>	書籍用
<code>&lt;report&gt;</code>	<code>jsreport.cls</code>	レポート用
<code>&lt;jspf&gt;</code>	<code>jspf.cls</code>	某学会誌用
<code>&lt;kiyou&gt;</code>	<code>kiyou.cls</code>	某紀要用

L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> あるいは pL<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> 標準のドキュメントクラスとの違いを説明しておきます。

■JIS フォントメトリックの使用 ここでは和文 TFM (T<sub>E</sub>X フォントメトリック) として東京書籍印刷の小林肇さんの作られた JIS フォントメトリック `jis.tfm`, `jisg.tfm` を標準で使います。従来のフォントメトリック `min10.tfm`, `goth10.tfm` の類を使うには

```
\documentclass[mingoth]{jsarticle}
```

のように `mingoth` オプションを付けます。

■サイズオプションの扱いが違う 標準のドキュメントクラスでは本文のポイント数を指定するオプションがありましたが、ポイント数は 10, 11, 12 しかなく、それぞれ別のクラスオプションファイルを読み込むようになっていました。しかも、標準の 10 ポイント以外では多少フォントのバランスが崩れることがあり、あまり便利ではありませんでした。ここでは文字サイズを増すとページを小さくし、 $\text{T}_{\text{E}}\text{X}$  の `\mag` プリミティブで全体的に拡大するという手を使って、9 ポイントや 21, 25, 30, 36, 43 ポイント, 12Q, 14Q の指定を可能にしています。

以下では実際のコードに即して説明します。

`minijs` は、`jsclasses` に似た設定を行うパッケージです。

```
1 <minijs>
2 %% if jsclasses loaded, abort loading this package
3 \ifx\@jsc@uplatextrue\@undefined\else
4   \PackageInfo{minijs}{jsclasses does not need minijs, exiting}
5   \expandafter\endinput
6 \fi
7 %% "fake" jsarticle
8 \expandafter\def\csname ver@jsarticle.cls\endcsname{}
9 </minijs>
```

`\jsc@clsname` 文書クラスの名前です。エラーメッセージ表示などで使われます。

```
10 <*class>
11 <article>\def\jsc@clsname{jsarticle}
12 <book>\def\jsc@clsname{jsbook}
13 <report>\def\jsc@clsname{jsreport}
14 <jspf>\def\jsc@clsname{jspf}
15 <kiyou>\def\jsc@clsname{kiyou}
```

`\ifjsc@needsp@tch` [2016-08-22] 従来 `jsclasses` では、 $\text{pL}_{\text{A}}\text{T}_{\text{E}}\text{X}$  や  $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$  の不都合な点に対して、クラスファイル内で独自に対策を施していました。しかし、2016 年以降、コミュニティ版  $\text{pL}_{\text{A}}\text{T}_{\text{E}}\text{X}$  が次第に対策コードをカーネル内に取り込むようになりました。そこで、新しい  $\text{pL}_{\text{A}}\text{T}_{\text{E}}\text{X}$  カーネルと衝突しないように、日付が古い場合だけパッチをあてる場合があります。この処理に使用するフラグを定義します。

```
16 \newif\ifjsc@needsp@tch
17 \jsc@needsp@tchfalse
```

## 2 オプション

これらのクラスは `\documentclass{jsarticle}` あるいは `\documentclass[オプション]{jsarticle}` のように呼び出します。

まず、オプションに関連するいくつかのコマンドやスイッチ（論理変数）を定義します。

`\if@restonecol` 段組のときに真になる論理変数です。

```
18 \newif\if@restonecol
```

`\if@titlepage` これを真にすると表題、概要を独立したページに出力します。  
19 `\newif\if@titlepage`

`\if@openright` `\chapter`, `\part` を右ページ起こしにするかどうかです。横組の書籍では真が標準で、要するに片起こし、奇数ページ起こしになります。  
20 `(book|report)\newif\if@openright`

`\if@openleft` [2017-02-24] `\chapter`, `\part` を左ページ起こしにするかどうかです。  
21 `(book|report)\newif\if@openleft`

`\if@mainmatter` 真なら本文、偽なら前付け・後付けです。偽なら `\chapter` で章番号が出ません。  
22 `(book)\newif\if@mainmatter \@mainmattertrue`

`\if@enablejfam` 和文フォントを数式フォントとして登録するかどうかを示すスイッチです。  
23 `\newif\if@enablejfam \@enablejfamtrue`

以下で各オプションを宣言します。

■用紙サイズ JIS や ISO の A0 判は面積  $1\text{ m}^2$ 、縦横比  $1:\sqrt{2}$  の長方形の辺の長さを mm 単位に切り捨てたものです。これを基準として順に半截しては mm 単位に切り捨てたものが A1, A2, …です。

B 判は JIS と ISO で定義が異なります。JIS では B0 判の面積が  $1.5\text{ m}^2$  ですが、ISO では B1 判の辺の長さが A0 判と A1 判の辺の長さの幾何平均です。したがって ISO の B0 判は  $1000\text{ mm} \times 1414\text{ mm}$  です。このため、 $\text{\LaTeX 2}_\epsilon$  の `b5paper` は  $250\text{ mm} \times 176\text{ mm}$  ですが、 $\text{p}\text{\LaTeX 2}_\epsilon$  の `b5paper` は  $257\text{ mm} \times 182\text{ mm}$  になっています。ここでは  $\text{p}\text{\LaTeX 2}_\epsilon$  にならって JIS に従いました。

デフォルトは `a4paper` です。

`b5var` (B5 変形,  $182\text{ mm} \times 230\text{ mm}$ ), `a4var` (A4 変形,  $210\text{ mm} \times 283\text{ mm}$ ) を追加しました。

```

24 \DeclareOption{a3paper}{%
25   \setlength\paperheight {420mm}%
26   \setlength\paperwidth  {297mm}}
27 \DeclareOption{a4paper}{%
28   \setlength\paperheight {297mm}%
29   \setlength\paperwidth  {210mm}}
30 \DeclareOption{a5paper}{%
31   \setlength\paperheight {210mm}%
32   \setlength\paperwidth  {148mm}}
33 \DeclareOption{a6paper}{%
34   \setlength\paperheight {148mm}%
35   \setlength\paperwidth  {105mm}}
36 \DeclareOption{b4paper}{%
37   \setlength\paperheight {364mm}%
38   \setlength\paperwidth  {257mm}}
39 \DeclareOption{b5paper}{%
40   \setlength\paperheight {257mm}%

```

```

41 \setlength\paperwidth {182mm}}
42 \DeclareOption{b6paper}{%
43 \setlength\paperheight {182mm}%
44 \setlength\paperwidth {128mm}}
45 \DeclareOption{a4j}{%
46 \setlength\paperheight {297mm}%
47 \setlength\paperwidth {210mm}}
48 \DeclareOption{a5j}{%
49 \setlength\paperheight {210mm}%
50 \setlength\paperwidth {148mm}}
51 \DeclareOption{b4j}{%
52 \setlength\paperheight {364mm}%
53 \setlength\paperwidth {257mm}}
54 \DeclareOption{b5j}{%
55 \setlength\paperheight {257mm}%
56 \setlength\paperwidth {182mm}}
57 \DeclareOption{a4var}{%
58 \setlength\paperheight {283mm}%
59 \setlength\paperwidth {210mm}}
60 \DeclareOption{b5var}{%
61 \setlength\paperheight {230mm}%
62 \setlength\paperwidth {182mm}}
63 \DeclareOption{letterpaper}{%
64 \setlength\paperheight {11in}%
65 \setlength\paperwidth {8.5in}}
66 \DeclareOption{legalpaper}{%
67 \setlength\paperheight {14in}%
68 \setlength\paperwidth {8.5in}}
69 \DeclareOption{executivepaper}{%
70 \setlength\paperheight {10.5in}%
71 \setlength\paperwidth {7.25in}}

```

■横置き 用紙の縦と横の長さを入れ換えます。

```

72 \newif\if@landscape
73 \@landscapefalse
74 \DeclareOption{landscape}{\@landscapetrue}

```

■slide オプション slide を新設しました。

[2016-10-08] slide オプションは article 以外では使い物にならなかったの、簡単のため article のみで使えるオプションとしました。

```

75 \newif\if@slide
76 \@slidefalse

```

■サイズオプション 10pt, 11pt, 12pt のほかに, 8pt, 9pt, 14pt, 17pt, 21pt, 25pt, 30pt, 36pt, 43pt を追加しました。これは等比数列になるように選んだものです (従来の 20pt も残しました)。`\@ptsize` の定義が変だったのでご迷惑をおかけしましたが、標準的なドキュメントクラスと同様にポイント数から 10 を引いたものに直しました。

[2003-03-22] 14Q オプションを追加しました。

[2003-04-18] 12Q オプションを追加しました。

[2016-07-08] `\mag` を使わずに各種寸法をスケールさせるためのオプション `nomag` を新設しました。`usemag` オプションの指定で従来通りの動作となります。デフォルトは `usemag` です。

[2016-07-24] オプティカルサイズを調整するために NFSS へパッチを当てるオプション `nomag*` を新設しました。

```
77 \newcommand{\@ptsize}{0}
78 \newif\ifjsc@mag\jsc@magtrue
79 \newif\ifjsc@mag@xreal\jsc@mag@xrealfalse
80 \def\jsc@magscale{1}
81 (*article)
82 \DeclareOption{slide}{%
83   \@slidetrue\def\jsc@magscale{3.583}
84   \renewcommand{\@ptsize}{26}
85   \@landscapetrue\@titlepagetrue}
86 \end{article}
87 \DeclareOption{8pt}{\def\jsc@magscale{0.833}\renewcommand{\@ptsize}{-2}}
88 \DeclareOption{9pt}{\def\jsc@magscale{0.913}\renewcommand{\@ptsize}{-1}}
89 \DeclareOption{10pt}{\def\jsc@magscale{1}\renewcommand{\@ptsize}{0}}
90 \DeclareOption{11pt}{\def\jsc@magscale{1.095}\renewcommand{\@ptsize}{1}}
91 \DeclareOption{12pt}{\def\jsc@magscale{1.200}\renewcommand{\@ptsize}{2}}
92 \DeclareOption{14pt}{\def\jsc@magscale{1.440}\renewcommand{\@ptsize}{4}}
93 \DeclareOption{17pt}{\def\jsc@magscale{1.728}\renewcommand{\@ptsize}{7}}
94 \DeclareOption{20pt}{\def\jsc@magscale{2}\renewcommand{\@ptsize}{10}}
95 \DeclareOption{21pt}{\def\jsc@magscale{2.074}\renewcommand{\@ptsize}{11}}
96 \DeclareOption{25pt}{\def\jsc@magscale{2.488}\renewcommand{\@ptsize}{15}}
97 \DeclareOption{30pt}{\def\jsc@magscale{2.986}\renewcommand{\@ptsize}{20}}
98 \DeclareOption{36pt}{\def\jsc@magscale{3.583}\renewcommand{\@ptsize}{26}}
99 \DeclareOption{43pt}{\def\jsc@magscale{4.300}\renewcommand{\@ptsize}{33}}
100 \DeclareOption{12Q}{\def\jsc@magscale{0.923}\renewcommand{\@ptsize}{1200}}
101 \DeclareOption{14Q}{\def\jsc@magscale{1.077}\renewcommand{\@ptsize}{1400}}
102 \DeclareOption{10ptj}{\def\jsc@magscale{1.085}\renewcommand{\@ptsize}{1001}}
103 \DeclareOption{10.5ptj}{\def\jsc@magscale{1.139}\renewcommand{\@ptsize}{1051}}
104 \DeclareOption{11ptj}{\def\jsc@magscale{1.194}\renewcommand{\@ptsize}{1101}}
105 \DeclareOption{12ptj}{\def\jsc@magscale{1.302}\renewcommand{\@ptsize}{1201}}
106 \DeclareOption{usemag}{\jsc@magtrue\jsc@mag@xrealfalse}
107 \DeclareOption{nomag}{\jsc@magfalse\jsc@mag@xrealfalse}
108 \DeclareOption{nomag*}{\jsc@magfalse\jsc@mag@xrealtrue}
```

■**トンボオプション** トンボ (crop marks) を出力します。実際の処理は `pLATEX 2ε` 本体で行います (`plcore.dtx` 参照)。オプション `tombow` で日付付きのトンボ、オプション `tombo` で日付なしのトンボを出力します。これらはアスキー版のままです。カウンタ `\hour`, `\minute` は `pLATEX 2ε` 本体で宣言されています。

```
109 \hour\time \divide\hour by 60\relax
110 \@tempcnta\hour \multiply\@tempcnta 60\relax
```

```

111 \minute\time \advance\minute-\@tempcnta
112 \DeclareOption{tombow}{%
113   \tombowtrue \tombowdatetrue
114   \setlength{\@tombowwidth}{.1\p}%
115   \@bannertoken{%
116     \jobname\space(\number\year-\two@digits\month-\two@digits\day
117     \space\two@digits\hour:\two@digits\minute)}}%
118   \maketombowbox}
119 \DeclareOption{tombo}{%
120   \tombowtrue \tombowdatefalse
121   \setlength{\@tombowwidth}{.1\p}%
122   \maketombowbox}

```

■面付け オプション `mentuke` で幅ゼロのトンボを出力します。面付けに便利です。これもアスキー版のままです。

```

123 \DeclareOption{mentuke}{%
124   \tombowtrue \tombowdatefalse
125   \setlength{\@tombowwidth}{.1\p}%
126   \maketombowbox}

```

■両面, 片面オプション `twoside` で奇数ページ・偶数ページのレイアウトが変わります。  
[2003-04-29] `vartwoside` でどちらのページも傍注が右側になります。

```

127 \DeclareOption{oneside}{\@twosidefalse \@mparswitchfalse}
128 \DeclareOption{twoside}{\@twosidetrue \@mparswitchtrue}
129 \DeclareOption{vartwoside}{\@twosidetrue \@mparswitchfalse}

```

■二段組 `twocolumn` で二段組になります。

```

130 \DeclareOption{onecolumn}{\@twocolumnfalse}
131 \DeclareOption{twocolumn}{\@twocolumntrue}

```

■表題ページ `titlepage` で表題・概要を独立したページに出力します。

```

132 \DeclareOption{titlepage}{\@titlepagetrue}
133 \DeclareOption{notitlepage}{\@titlepagefalse}

```

■右左起こし 書籍では章は通常は奇数ページ起こしになりますが、横組ではこれを `openright` と表すことにしてあります。 `openany` で偶数ページからでも始まるようになります。

[2017-02-24] `openright` は横組では奇数ページ起こし、縦組では偶数ページ起こしを表します。ややこしいですが、これは  $\text{\LaTeX}$  の標準クラスが西欧の横組事情しか考慮せずに、奇数ページ起こしと右起こしを一緒にしてしまったせいです。縦組での奇数ページ起こしと横組での偶数ページ起こしも表現したいので、`jsclasses` では新たに `openleft` も追加しました。

```

134 \book | report \DeclareOption{openright}{\@openrighttrue\@openleftfalse}
135 \book | report \DeclareOption{openleft}{\@openlefttrue\@openrightfalse}
136 \book | report \DeclareOption{openany}{\@openrightfalse\@openleftfalse}

```

■eqnarray 環境と数式の位置 森本さんのご教示にしたがって前に移動しました。

eqnarray L<sup>A</sup>T<sub>E</sub>X の eqnarray 環境では & でできるアキが大きすぎるようですので、少し小さくします。また、中央の要素も \displaystyle にします。

```

137 \def\eqnarray{%
138   \stepcounter{equation}%
139   \def\@currentlabel{\p@equation\theequation}%
140   \global\@eqnswtrue
141   \m@th
142   \global\@eqcnt\z@
143   \tabskip\@centering
144   \let\@=\@eqncr
145   $$\everycr{}\halign to\displaywidth\bgroup
146     \hskip\@centering$\displaystyle\tabskip\z@skip{##}$\@eqnse1
147     &\global\@eqcnt\@ne \hfil$\displaystyle{##}$\hfil
148     &\global\@eqcnt\tw@ $\displaystyle{##}$\hfil\tabskip\@centering
149     &\global\@eqcnt\thr@@ \hb@xt@\z@\bgroup\hss##\egroup
150     \tabskip\z@skip
151   \cr}

```

leqno で数式番号が左側になります。fleqn で数式が本文左端から一定距離のところに出力されます。森本さんにしたがって訂正しました。

```

152 \DeclareOption{leqno}{\input{leqno.clo}}
153 \DeclareOption{fleqn}{\input{fleqn.clo}}
154 % fleqn 用の eqnarray 環境の再定義
155 \def\eqnarray{%
156   \stepcounter{equation}%
157   \def\@currentlabel{\p@equation\theequation}%
158   \global\@eqnswtrue\m@th
159   \global\@eqcnt\z@
160   \tabskip\mathindent
161   \let\@=\@eqncr
162   \setlength\abovedisplayskip{\topsep}%
163   \ifvmode
164     \addtolength\abovedisplayskip{\partopsep}%
165   \fi
166   \addtolength\abovedisplayskip{\parskip}%
167   \setlength\belowdisplayskip{\abovedisplayskip}%
168   \setlength\belowdisplayshortskip{\abovedisplayskip}%
169   \setlength\abovedisplayshortskip{\abovedisplayskip}%
170   $$\everycr{}\halign to\linewidth% $$
171   \bgroup
172     \hskip\@centering$\displaystyle\tabskip\z@skip{##}$\@eqnse1
173     &\global\@eqcnt\@ne \hfil$\displaystyle{##}$\hfil
174     &\global\@eqcnt\tw@
175     $\displaystyle{##}$\hfil \tabskip\@centering
176     &\global\@eqcnt\thr@@ \hb@xt@\z@\bgroup\hss##\egroup
177   \tabskip\z@skip\cr

```

178       }}}

■文献リスト 文献リストを open 形式（著者名や書名の後に改行が入る）で出力します。  
これは使われることはないのでコメントアウトしてあります。

```
179 % \DeclareOption{openbib}{%
180 %   \AtEndOfPackage{%
181 %     \renewcommand\@openbib@code{%
182 %       \advance\leftmargin\bibindent
183 %       \itemindent -\bibindent
184 %       \listparindent \itemindent
185 %       \parsep \z@}%
186 %     \renewcommand\newblock{\par}}}
```

■数式フォントとして和文フォントを登録しないオプション 数式中では 16 通りのフォントしか使えません。AMSFonTS や mathptmx パッケージを使って数式フォントをたくさん使うと “Too many math alphabets ...” というエラーが起こってしまいます。disablejfam オプションを付ければ、明朝・ゴシックを数式用フォントとして登録するのをやめますので、数式用フォントが二つ節約できます。いずれにしても \textmc や \mbox や amsmath パッケージの \text を使えば数式中で和文フォントが使えますので、この新ドキュメントクラスでは標準で和文フォントを数式用に登録しないことにしていたのですが、従来のドキュメントクラスの仕様に合わせることにしました。

```
187 \DeclareOption{disablejfam}{\@enablejfamfalse}
```

■ドラフト draft で overfull box の起きた行末に 5pt の罫線を引きます。

[2016-07-13] \ifdraft を定義するのをやめました。

```
188 \DeclareOption{draft}{\setlength\overfullrule{5pt}}
189 \DeclareOption{final}{\setlength\overfullrule{0pt}}
```

■和文フォントメトリックの選択 このクラスファイルでは、和文 TFM として東京書籍印刷の小林肇さんの作られた JIS フォントメトリック (jis, jisg) を標準で使うことにしますが、従来の min10, goth10 などを使いたいときは mingoth というオプションを指定します。また、winjis オプションで winjis メトリック (OTF パッケージと同じ psitau さん作；ソースに書かれた Windows の機種依存文字が dvips, dvipdfmx などでも出力出来るようになる) が使えます。

[2018-02-04] winjis オプションはコッソリ削除しました。代替として、同等なものをパッケージ化 (winjis.sty) して、GitHub にはコッソリ置いておきます。

```
190 \newif\ifmingoth
191 \mingothfalse
192 \newif\ifjisfont
193 \jisfontfalse
194 \newif\if@jsc@uplatex
195 \@jsc@uplatexfalse
196 \newif\if@jsc@autodetect
197 \@jsc@autodetectfalse
```



```

198 \DeclareOption{winjis}{%
199   \ClassWarningNoLine{\jsc@clsname}{%
200     The option ‘winjis’ has been removed;\MessageBreak
201     Use ‘\string\usepackage{winjis}’ instead}}
202 \DeclareOption{mingoth}{\mingothtrue}
203 \DeclareOption{jis}{\jisfonttrue}
204 \DeclareOption{uplatex}{\@jsc@uplatextrue}
205 \DeclareOption{autodetect-engine}{\@jsc@autodetecttrue}
206 \def\jsc@JYn{\if@jsc@uplatex JY2\else JY1\fi}
207 \def\jsc@JTn{\if@jsc@uplatex JT2\else JT1\fi}
208 \def\jsc@pfx@{\if@jsc@uplatex u\else \fi}

```

■papersize スペシャルの利用 dvips や dviout で用紙設定を自動化するにはオプション papersize を与えます。

```

209 \newif\ifpapersize
210 \papersizefalse
211 \DeclareOption{papersize}{\papersizetrue}

```

■英語化 オプション english を新設しました。

```

212 \newif\if@english
213 \@englishfalse
214 \DeclareOption{english}{\@englishttrue}

```

■jsbook を jsreport もどきに オプション report を新設しました。

[2017-02-13] 従来は「jsreport 相当」を jsbook の report オプションで提供していましたが、新しく jsreport クラスも作りました。どちらでも好きな方を使ってください。

```

215 (*book)
216 \newif\if@report
217 \@reportfalse
218 \DeclareOption{report}{\@reporttrue\@openrightfalse\@twosidefalse\@mparswitchfalse}
219 \book

```

■jslogo パッケージの読み込み L<sup>A</sup>T<sub>E</sub>X 関連のロゴを再定義する jslogo パッケージを読み込まないオプション nojslogo を新設しました。jslogo オプションの指定で従来どおりの動作となります。デフォルトは jslogo で、すなわちパッケージを読み込みます。

```

220 \newif\if@jslogo \@jslogotrue
221 \DeclareOption{jslogo}{\@jslogotrue}
222 \DeclareOption{nojslogo}{\@jslogofalse}

```

■オプションの実行 デフォルトのオプションを実行します。multicols や url を \RequirePackage するのはやめました。

```

223 \article\ExecuteOptions{a4paper,oneside,onecolumn,notitlepage,final}
224 \book\ExecuteOptions{a4paper,twoside,onecolumn,titlepage,openright,final}
225 \report\ExecuteOptions{a4paper,oneside,onecolumn,titlepage,openany,final}
226 \jspf\ExecuteOptions{a4paper,twoside,twocolumn,notitlepage,fleqn,final}
227 \kiyou\ExecuteOptions{a4paper,twoside,twocolumn,notitlepage,final}

```

```
228 \ProcessOptions
```

後処理

```
229 \if@slide
```

```
230 \def\maybeblue{\ifundefined{ver@color.sty}{\color{blue}}}
```

```
231 \fi
```

```
232 \if@landscape
```

```
233 \setlength\@tempdima {\paperheight}
```

```
234 \setlength\paperheight{\paperwidth}
```

```
235 \setlength\paperwidth {\@tempdima}
```

```
236 \fi
```

■使用エンジンの検査・自動判定 ユーザが `uplatex` オプションの有無により指定したエンジンが、実際に使われているものと一致しているかを検査し、一致しない場合はエラーメッセージを表示します。

[2016-11-09] `pLATEX` / `upLATEX` を自動判別するオプション `autodetect-engine` を新設しました。`upLATEX` の場合は、グローバルオプションに `uplatex` を追加することで、自動判定に応じて `otf` パッケージにも `uplatex` オプションが渡るようにします。

```
237 \ifnum \ifx\ucs\@undefined\z@\else\ucs"3000 \fi ="3000
```

```
238 \if@jsc@autodetect
```

```
239 \ClassInfo\jsc@clsname{Autodetected engine: upLaTeX}
```

```
240 \@jsc@uplatextrue
```

```
241 \g@addto@macro\@classoptionslist{,uplatex}
```

```
242 \fi
```

```
243 \if@jsc@uplatex\else
```

```
244 \ClassError\jsc@clsname
```

```
245 {You are running upLaTeX.\MessageBreak
```

```
246 Please use pLaTeX instead, or add 'uplatex' to\MessageBreak
```

```
247 the class option list}
```

```
248 {\@ehc}
```

```
249 \@jsc@uplatextrue
```

```
250 \fi
```

[2016-11-11] `pLATEX` の場合は、オプション `uplatex` が指定されていれば必ずエラーを出します。`autodetect-engine` が有効になっていてもエラーを出しますが、これは `otf` パッケージに `uplatex` オプションが渡ってしまうのを防ぐためです。

```
251 \else
```

```
252 \if@jsc@uplatex
```

```
253 \ClassError\jsc@clsname
```

```
254 {You are running pLaTeX.\MessageBreak
```

```
255 Please use upLaTeX instead, or remove 'uplatex' from\MessageBreak
```

```
256 the class option list}
```

```
257 {\@ehc}
```

```
258 \@jsc@uplatexfalse
```

```
259 \fi
```

```
260 \if@jsc@autodetect
```

```
261 \ClassInfo\jsc@clsname{Autodetected engine: pLaTeX}
```

```
262 \@jsc@uplatexfalse
```

```
263 \fi
264 \fi
```

■papersize スペシャルの出力 dvi ファイルの先頭に dvips の papersize special を書き込むことで、出力用紙サイズを設定します。これは dvipdfmx や最近の dviout にも有効です。どうやら papersize special には true 付の単位は許されず、かつ単位は常に true なものと扱われるようです。そこで、後で出てくる（☆）の部分、「\mag にあわせてスケール」よりも手前で実行しておくことになります。

トンボの付いたときの用紙サイズは無意味ですが、いわゆる「ノビ」サイズという縦横 1 インチずつ長い用紙に出力することを考えて、1 インチずつ加えました。ところが pL<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> はトンボ出力幅を両側に 1 インチとっていますので、dvips 使用時に

```
-0 -0.5in,-0.5in
```

というオプションを与えて両側 0.5 インチのトンボにするといいでしょう。

[2003-05-17] トンボをプレビューに使うことを考えて 1 インチを 2 インチにしました。

[2016-07-11] memoir クラスのマニュアルによると、トンボを含めた用紙の寸法は \stockwidth, \stockheight と呼ぶようですので、これを使うことにしました。

[2017-01-11] トンボオプションが指定されているとき「だけ」\stockwidth, \stockheight を定義するようにしました。

[2020-10-04] L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> 2020-10-01 でカーネルの \shipout コードが拡張され \AtBeginDvi の実行タイミングが変化したので、この時点で発行する \special の中身を展開しておくようにしました。こうしないと、用紙サイズ設定を間違ってしまう (Issue #72)。

```
265 \iftombow
266 \newdimen\stockwidth \newdimen\stockheight
267 \setlength{\stockwidth}{\paperwidth}
268 \setlength{\stockheight}{\paperheight}
269 \advance \stockwidth 2in
270 \advance \stockheight 2in
271 \fi
272 \ifpapersize
273 \iftombow
274 \edef\jsc@papersize@special{papersize=\the\stockwidth,\the\stockheight}
275 \else
276 \edef\jsc@papersize@special{papersize=\the\paperwidth,\the\paperheight}
277 \fi
278 \AtBeginDvi{\special{\jsc@papersize@special}}
279 \fi
```

## ■基準となる行送り

\n@baseline 基準となる行送りをポイント単位で表したものです。

```
280 <article|book|report>\if@slide\def\n@baseline{13}\else\def\n@baseline{16}\fi
281 <jspf>\def\n@baseline{14.554375}
```

```
282 \kiyou\def\n@baseline{14.897}
```

■**拡大率の設定** サイズの変更は T<sub>E</sub>X のプリミティブ `\mag` を使って行います。9 ポイントについては行送りも若干縮めました。サイズについては全面的に見直しました。

[2008-12-26] 1000 / `\mag` に相当する `\inv@mag` を定義しました。`truein` を使っていたところを `\inv@mag in` に直しましたので、`geometry` パッケージと共存できると思います。なお、新ドキュメントクラス側で 10pt 以外にする場合の注意：

- `geometry` 側でオプション `truedimen` を指定してください。
- `geometry` 側でオプション `mag` は使えません。

[2016-07-08] `\jsc@mp` および `\jsc@mm` に、それぞれ 1pt および 1mm を拡大させた値を格納します。以降のレイアウト指定ではこちらを使います。

```
283 \newdimen\jsc@mp
284 \newdimen\jsc@mm
285 \def\inv@mag{1}
286 \ifjsc@mag
287   \jsc@mp=1\p@
288   \jsc@mm=1mm
289   \ifnum\@ptsize=-2
290     \mag 833
291     \def\inv@mag{1.20048}
292     \def\n@baseline{15}%
293   \fi
294   \ifnum\@ptsize=-1
295     \mag 913 % formerly 900
296     \def\inv@mag{1.09529}
297     \def\n@baseline{15}%
298   \fi
299   \ifnum\@ptsize=1
300     \mag 1095 % formerly 1100
301     \def\inv@mag{0.913242}
302   \fi
303   \ifnum\@ptsize=2
304     \mag 1200
305     \def\inv@mag{0.833333}
306   \fi
307   \ifnum\@ptsize=4
308     \mag 1440
309     \def\inv@mag{0.694444}
310   \fi
311   \ifnum\@ptsize=7
312     \mag 1728
313     \def\inv@mag{0.578704}
314   \fi
315   \ifnum\@ptsize=10
316     \mag 2000
```

```

317     \def\inv@mag{0.5}
318 \fi
319 \ifnum\@ptsize=11
320     \mag 2074
321     \def\inv@mag{0.48216}
322 \fi
323 \ifnum\@ptsize=15
324     \mag 2488
325     \def\inv@mag{0.401929}
326 \fi
327 \ifnum\@ptsize=20
328     \mag 2986
329     \def\inv@mag{0.334896}
330 \fi
331 \ifnum\@ptsize=26
332     \mag 3583
333     \def\inv@mag{0.279096}
334 \fi
335 \ifnum\@ptsize=33
336     \mag 4300
337     \def\inv@mag{0.232558}
338 \fi
339 \ifnum\@ptsize=1200
340     \mag 923
341     \def\inv@mag{1.0834236}
342 \fi
343 \ifnum\@ptsize=1400
344     \mag 1077
345     \def\inv@mag{0.928505}
346 \fi
347 \ifnum\@ptsize=1001
348     \mag 1085
349     \def\inv@mag{0.921659}
350 \fi
351 \ifnum\@ptsize=1051
352     \mag 1139
353     \def\inv@mag{0.877963}
354 \fi
355 \ifnum\@ptsize=1101
356     \mag 1194
357     \def\inv@mag{0.837521}
358 \fi
359 \ifnum\@ptsize=1201
360     \mag 1302
361     \def\inv@mag{0.768049}
362 \fi
363 \else
364     \jsc@empt=\jsc@magscale\p@
365     \jsc@mmm=\jsc@magscale mm

```

```

366 \def\inv@mag{1}
367 \ifnum\@ptsize=-2
368 \def\n@baseline{15}%
369 \fi
370 \ifnum\@ptsize=-1
371 \def\n@baseline{15}%
372 \fi
373 \fi
374 <*kiyou>
375 \def\jsc@magscale{0.9769230}
376 \ifjsc@mag
377 \mag 977
378 \def\inv@mag{1.02354}
379 \jsc@mpt=1\p@
380 \jsc@mmm=1mm
381 \else
382 \jsc@mpt=\jsc@magscale\p@
383 \jsc@mmm=\jsc@magscale mm
384 \def\inv@mag{1}
385 \fi
386 </kiyou>
387 \ifjsc@mag@xreal
388 \RequirePackage{type1cm}
389 \mathchardef\jsc@csta=259
390 \def\jsc@invscale#1#2{%
391 \begingroup \@tempdima=#1\relax \@tempdimb#2\p@\relax
392 \@tempcnta\@tempdima \multiply\@tempcnta\@ccclvi
393 \divide\@tempcnta\@tempdimb \multiply\@tempcnta\@ccclvi
394 \@tempcntb\p@ \divide\@tempcntb\@tempdimb
395 \advance\@tempcnta-\@tempcntb \advance\@tempcnta-\tw@
396 \@tempdimb\@tempcnta\@ne
397 \advance\@tempcnta\@tempcntb \advance\@tempcnta\@tempcntb
398 \advance\@tempcnta\jsc@csta \@tempdimc\@tempcnta\@ne
399 \@whiledim\@tempdimb<\@tempdimc\do{%
400 \@tempcntb\@tempdimb \advance\@tempcntb\@tempdimc
401 \advance\@tempcntb\@ne \divide\@tempcntb\tw@
402 \ifdim #2\@tempcntb>\@tempdima
403 \advance\@tempcntb\m@ne \@tempdimc=\@tempcntb\@ne
404 \else \@tempdimb=\@tempcntb\@ne \fi}%
405 \xdef\jsc@gtmpa{\the\@tempdimb}%
406 \endgroup #1=\jsc@gtmpa\relax}
407 \expandafter\let\csname OT1/cmr/m/n/10\endcsname\relax
408 \expandafter\let\csname OMX/cmex/m/n/10\endcsname\relax
409 \let\jsc@get@external@font\get@external@font
410 \def\get@external@font{%
411 \jsc@preadjust@extract@font
412 \jsc@get@external@font}
413 \def\jsc@fstrunc#1{%
414 \edef\jsc@tmpa{\strip@pt#1}%

```

```

415 \expandafter\jsc@fstrunc@a\jsc@tmpa.****\@nil}
416 \def\jsc@fstrunc@a#1.#2#3#4#5#6\@nil{%
417 \if#5*\else
418 \edef\jsc@tmpa{#1%
419 \ifnum#2#3>\z@ .#2\ifnum#3>\z@ #3\fi\fi}%
420 \fi}
421 \def\jsc@preadjust@extract@font{%
422 \let\jsc@req@size\f@size
423 \dimen@f@size\p@ \jsc@invscale\dimen@\jsc@magscale
424 \advance\dimen@.005pt\relax \jsc@fstrunc\dimen@
425 \let\jsc@ref@size\jsc@tmpa
426 \let\f@size\jsc@ref@size}
427 \def\execute@size@function#1{%
428 \let\jsc@cref@size\f@size
429 \let\f@size\jsc@req@size
430 \csname s@fct@#1\endcsname}
431 \let\jsc@DeclareErrorFont\DeclareErrorFont
432 \def\DeclareErrorFont#1#2#3#4#5{%
433 \@tempdimc#5\p@ \@tempdimc\jsc@magscale\@tempdimc
434 \edef\jsc@tmpa{{#1}-{#2}-{#3}-{#4}-{strip@pt\@tempdimc}}
435 \expandafter\jsc@DeclareErrorFont\jsc@tmpa}
436 \def\gen@sfcnt{%
437 \edef\mandatory@arg{\mandatory@arg\jsc@cref@size}%
438 \empty@sfcnt}
439 \def\genb@sfcnt{%
440 \edef\mandatory@arg{%
441 \mandatory@arg\expandafter\genb@x\jsc@cref@size..\@@}%
442 \empty@sfcnt}
443 \DeclareErrorFont{OT1}{cmr}{m}{n}{10}
444 \fi

```

[2016-11-16] latex.ltx (ltspace.dtx) で定義されている `\smallskip` の、単位 `pt` を `\jsc@mpt` に置き換えた `\jsc@smallskip` を定義します。これは `\maketitle` で用いられます。`\jsc@medskip` と `\jsc@bigskip` は必要ないのでコメントアウトしています。

`\jsc@smallskip`

```

\jsc@medskip 445 \def\jsc@smallskip{\vspace\jsc@smallskipamount}
\jsc@bigskip 446 %\def\jsc@medskip{\vspace\jsc@medskipamount}
447 %\def\jsc@bigskip{\vspace\jsc@bigskipamount}

```

`\jsc@smallskipamount`

```

\jsc@medskipamount 448 \newskip\jsc@smallskipamount
\jsc@bigskipamount 449 \jsc@smallskipamount=3\jsc@mpt plus 1\jsc@mpt minus 1\jsc@mpt
450 %\newskip\jsc@medskipamount
451 %\jsc@medskipamount =6\jsc@mpt plus 2\jsc@mpt minus 2\jsc@mpt
452 %\newskip\jsc@bigskipamount
453 %\jsc@bigskipamoun =12\jsc@mpt plus 4\jsc@mpt minus 4\jsc@mpt

```

`\paperwidth`, `\paperheight` を `\mag` にあわせてスケールしておきます (☆)。

[2016-07-11] 新しく追加した`\stockwidth`, `\stockheight` も`\mag`にあわせてスケールします。

[2017-01-11] トンボオブションが指定されているとき「だけ」`\stockwidth`, `\stockheight` が定義されています。

```
454 \setlength\paperwidth{\inv@mag\paperwidth}%
455 \setlength\paperheight{\inv@mag\paperheight}%
456 \iftombow
457   \setlength\stockwidth{\inv@mag\stockwidth}%
458   \setlength\stockheight{\inv@mag\stockheight}%
459 \fi
```

■**pagesize スペシャルの出力** [2003-05-17] `dvipdfm(x)` の `pagesize` スペシャルを出力します。

[2004-08-08] 今の `dvipdfmx` は `dvips` 用スペシャルを理解するようなので外しました。

```
460 % \ifpapersize
461 %   \setlength{\@tempdima}{\paperwidth}
462 %   \setlength{\@tempdimb}{\paperheight}
463 %   \iftombow
464 %     \advance \@tempdima 2truein
465 %     \advance \@tempdimb 2truein
466 %   \fi
467 %   \AtBeginDvi{\special{pdf: pagesize width \the\@tempdima space height \the\@tempdimb}}
468 % \fi
```

### 3 和文フォントの変更

JIS の 1 ポイントは 0.3514mm (約 1/72.28 インチ), PostScript の 1 ポイントは 1/72 インチですが,  $\mathrm{T}_{\mathrm{E}}\mathrm{X}$  では 1/72.27 インチを 1pt (ポイント), 1/72 インチを 1bp (ビッグポイント) と表します。QuarkXPress などの DTP ソフトは標準で 1/72 インチを 1 ポイントとしますが, 以下ではすべて 1/72.27 インチを 1pt としています。1 インチは定義により 25.4mm です。

さらにややこしいことに,  $\mathrm{pT}_{\mathrm{E}}\mathrm{X}$  (アスキーが日本語化した  $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ ) の公称 10 ポイントの和文フォント (`min10` など) は, 実寸 (標準の字送り量) が 9.62216pt です。これは 3.3818mm, 写研の写植機の単位では 13.527 級, PostScript の単位では 9.5862 ポイントになります。`jis` フォントなどもこの値を踏襲しています。

この公称 10 ポイントのフォントを, ここでは 13 級に縮小して使うことにします。そのためには,  $13/13.527 = 0.961$  倍すればいいことになります (`min10` や `jis` の場合)。9.62216 ポイントの和文フォントをさらに 0.961 倍したことにより, 約 9.25 ポイント, DTP で使う単位 (1/72 インチ) では 9.21 ポイントということになり, 公称 10 ポイントといっても実は 9 ポイント強になります。

[2018-02-04] 上記のと通りの「クラスファイルが意図する和文スケール値 (1zw ÷ 要求サイズ)」を表す実数値マクロ `\Cjascalse` を定義します。このマクロが定義されている場合, OTF パッケージ (2018/02/01 以降のバージョン) はこれに従います。`jsarticle`,



jsbook, jsreport では,  $9.62216 \text{ pt} * 0.961 / 10 \text{ pt} = 0.924690$  です。

```

469 </class>
470 <*minijs>
471 %% min/goth -> jis/jisg (for pLaTeX only)
472 \ifx\ucs\@undefined
473 \@for\@tempa:=5,6,7,8,9,10,10.95,12,14.4,17.28,20.74,24.88\do{%
474   \expandafter\let\csname JY1/mc/m/n/\@tempa\endcsname\relax
475   \expandafter\let\csname JY1/gt/m/n/\@tempa\endcsname\relax
476   \expandafter\let\csname JT1/mc/m/n/\@tempa\endcsname\relax
477   \expandafter\let\csname JT1/gt/m/n/\@tempa\endcsname\relax
478 }
479 \def\Cjascale{0.924690}
480 \DeclareFontShape{JY1}{mc}{m}{n}{<-> s * [0.961] jis}{f}
481 \DeclareFontShape{JY1}{gt}{m}{n}{<-> s * [0.961] jisg}{f}
482 \DeclareFontShape{JT1}{mc}{m}{n}{<-> s * [0.961] tmin10}{f}
483 \DeclareFontShape{JT1}{gt}{m}{n}{<-> s * [0.961] tgoth10}{f}
484 \fi
485 </minijs>
486 <*class>
487 <!*jspf>
488 \def\Cjascale{0.924690}
489 \ifmingoth
490   \DeclareFontShape{jsc@JYn}{mc}{m}{n}{<-> s * [0.961] jsc@pfx@ min10}{f}
491   \DeclareFontShape{jsc@JYn}{gt}{m}{n}{<-> s * [0.961] jsc@pfx@ goth10}{f}
492   \DeclareFontShape{jsc@JTn}{mc}{m}{n}{<-> s * [0.961] jsc@pfx@ tmin10}{f}
493   \DeclareFontShape{jsc@JTn}{gt}{m}{n}{<-> s * [0.961] jsc@pfx@ tgoth10}{f}
494 \else
495   \ifjisfont
496     \DeclareFontShape{jsc@JYn}{mc}{m}{n}{<-> s * [0.961] jsc@pfx@ jis}{f}
497     \DeclareFontShape{jsc@JYn}{gt}{m}{n}{<-> s * [0.961] jsc@pfx@ jisg}{f}
498     \DeclareFontShape{jsc@JTn}{mc}{m}{n}{<-> s * [0.961] jsc@pfx@ tmin10}{f}
499     \DeclareFontShape{jsc@JTn}{gt}{m}{n}{<-> s * [0.961] jsc@pfx@ tgoth10}{f}
500   \else
501     \if@jsc@uplatex
502       \DeclareFontShape{JY2}{mc}{m}{n}{<-> s * [0.924690] upjisr-h}{f}
503       \DeclareFontShape{JY2}{gt}{m}{n}{<-> s * [0.924690] upjisg-h}{f}
504       \DeclareFontShape{JT2}{mc}{m}{n}{<-> s * [0.924690] upjisr-v}{f}
505       \DeclareFontShape{JT2}{gt}{m}{n}{<-> s * [0.924690] upjisg-v}{f}
506     \else
507       \DeclareFontShape{jsc@JYn}{mc}{m}{n}{<-> s * [0.961] jsc@pfx@ jis}{f}
508       \DeclareFontShape{jsc@JYn}{gt}{m}{n}{<-> s * [0.961] jsc@pfx@ jisg}{f}
509       \DeclareFontShape{jsc@JTn}{mc}{m}{n}{<-> s * [0.961] jsc@pfx@ tmin10}{f}
510       \DeclareFontShape{jsc@JTn}{gt}{m}{n}{<-> s * [0.961] jsc@pfx@ tgoth10}{f}
511     \fi
512   \fi
513 \fi
514 </!jspf>

```

某学会誌では, 和文フォントを PostScript の 9 ポイントにするために,  $9 / (9.62216 * 10)$

72/72.27) = 0.93885 倍します。

[2018-02-04] 和文スケール値 \Cjascale は  $9.62216 \text{ pt} * 0.93885 / 10 \text{ pt} = 0.903375$  です。

```
515 \*jspf
516 \def\Cjascale{0.903375}
517 \ifmingoth
518 \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ min10}{ }
519 \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ goth10}{ }
520 \DeclareFontShape{\jsc@JTn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tmin10}{ }
521 \DeclareFontShape{\jsc@JTn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tgoth10}{ }
522 \else
523 \ifjisfont
524 \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ jis}{ }
525 \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ jisg}{ }
526 \DeclareFontShape{\jsc@JTn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tmin10}{ }
527 \DeclareFontShape{\jsc@JTn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tgoth10}{ }
528 \else
529 \if@jsc@uplatex
530 \DeclareFontShape{JY2}{mc}{m}{n}{<-> s * [0.903375] upjisr-h}{ }
531 \DeclareFontShape{JY2}{gt}{m}{n}{<-> s * [0.903375] upjisg-h}{ }
532 \DeclareFontShape{JT2}{mc}{m}{n}{<-> s * [0.903375] upjisr-v}{ }
533 \DeclareFontShape{JT2}{gt}{m}{n}{<-> s * [0.903375] upjisg-v}{ }
534 \else
535 \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ jis}{ }
536 \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ jisg}{ }
537 \DeclareFontShape{\jsc@JTn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tmin10}{ }
538 \DeclareFontShape{\jsc@JTn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tgoth10}{ }
539 \fi
540 \fi
541 \fi
542 \end{jspf}
```

和文でイタリック体、斜体、サンセリフ体、タイプライタ体の代わりにゴシック体を使うことにします。

[2003-03-16] イタリック体、斜体について、和文でゴシックを当てていましたが、数学の定理環境などで多量のイタリック体を使うことがあり、ゴシックにすると黒々となってしまうという弊害がありました。amsthm を使わない場合は定理の本文が明朝になるように \newtheorem 環境を手直ししてしのいでいましたが、 $\text{T}_{\text{E}}\text{X}$  が数学で多用されることを考えると、イタリック体に明朝体を当てたほうがいいように思えてきましたので、イタリック体・斜体に対応する和文を明朝体に変えることにしました。

[2004-11-03] \rmfamily も和文対応にしました。

```
543 % \DeclareFontShape{\jsc@JYn}{mc}{bx}{n}{<->ssub*gt/m/n}{ } % in \jsc@JYnmc
544 % \DeclareFontShape{\jsc@JYn}{gt}{bx}{n}{<->ssub*gt/m/n}{ } % in \jsc@JYngt
545 \DeclareFontShape{\jsc@JYn}{mc}{m}{it}{<->ssub*mc/m/n}{ }
546 \DeclareFontShape{\jsc@JYn}{mc}{m}{sl}{<->ssub*mc/m/n}{ }
547 \DeclareFontShape{\jsc@JYn}{mc}{m}{sc}{<->ssub*mc/m/n}{ }
548 \DeclareFontShape{\jsc@JYn}{gt}{m}{it}{<->ssub*gt/m/n}{ }
549 \DeclareFontShape{\jsc@JYn}{gt}{m}{sl}{<->ssub*gt/m/n}{ }
```

```

550 \DeclareFontShape{\jsc@JYn}{mc}{bx}{it}{<->ssub*gt/m/n}{ }
551 \DeclareFontShape{\jsc@JYn}{mc}{bx}{sl}{<->ssub*gt/m/n}{ }
552 % \DeclareFontShape{\jsc@JTn}{mc}{bx}{n}{<->ssub*gt/m/n}{ } % in \jsc@JTnmc
553 % \DeclareFontShape{\jsc@JTn}{gt}{bx}{n}{<->ssub*gt/m/n}{ } % in \jsc@JTngt
554 \DeclareFontShape{\jsc@JTn}{mc}{m}{it}{<->ssub*mc/m/n}{ }
555 \DeclareFontShape{\jsc@JTn}{mc}{m}{sl}{<->ssub*mc/m/n}{ }
556 \DeclareFontShape{\jsc@JTn}{mc}{m}{sc}{<->ssub*mc/m/n}{ }
557 \DeclareFontShape{\jsc@JTn}{gt}{m}{it}{<->ssub*gt/m/n}{ }
558 \DeclareFontShape{\jsc@JTn}{gt}{m}{sl}{<->ssub*gt/m/n}{ }
559 \DeclareFontShape{\jsc@JTn}{mc}{bx}{it}{<->ssub*gt/m/n}{ }
560 \DeclareFontShape{\jsc@JTn}{mc}{bx}{sl}{<->ssub*gt/m/n}{ }

```

[2020-02-02] L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> 2020-02-02 で NFSS が拡張され、それに伴いオリジナルの `\rmfamily` などの定義が変化しました。`\DeclareRobustCommand` で直接定義すると、これを上書きして NFSS の拡張部分を壊してしまいますので、新たに提供されたフックにコードを挿入します。従来のコードも L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> 2019-10-01 以前のために残してありますが、`mweights` パッケージ対策も施しました (forum:2763)。

[2020-10-04] L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> 2020-10-01 では `\AddToHook` を利用します。

```

561 </class>
562 <*class | minijs>
563 %% ad-hoc "relation font"
564 \@ifl@t@r\fmtversion{2020/10/01}
565     {\jsc@needspace@tchfalse}{\jsc@needspace@tchtrue}
566 \ifjsc@needspace@tch           % --- for 2020-02-02 or older BEGIN
567 \ifx\@rmfamilyhook\@undefined % old
568 \DeclareRobustCommand\rmfamily
569     {\not@math@alphabet\rmfamily\mathrm}
570     \romanfamily\rmdefault\kanjifamily\mcdefault\selectfont}
571 \DeclareRobustCommand\sffamily
572     {\not@math@alphabet\sffamily\mathsf}
573     \romanfamily\sfdefault\kanjifamily\gtdefault\selectfont}
574 \DeclareRobustCommand\ttfamily
575     {\not@math@alphabet\ttfamily\mathtt}
576     \romanfamily\ttdefault\kanjifamily\gtdefault\selectfont}
577 \AtBeginDocument{%
578     \ifx\mweights@init\@undefined\else % mweights.sty is loaded
579         % my definitions above should have been overwritten, recover it!
580         % \selectfont is executed twice but I don't care about speed...
581         \expandafter\g@addto@macro\csname rmfamily \endcsname
582             {\kanjifamily\mcdefault\selectfont}%
583         \expandafter\g@addto@macro\csname sffamily \endcsname
584             {\kanjifamily\gtdefault\selectfont}%
585         \expandafter\g@addto@macro\csname ttfamily \endcsname
586             {\kanjifamily\gtdefault\selectfont}%
587     \fi}
588 \else % 2020-02-02
589 \g@addto@macro\@rmfamilyhook
590     {\prepare@family@series@update@kanji{mc}\mcdefault}

```

```

591 \g@addto@macro\@sffamilyhook
592 {\prepare@family@series@update@kanji{gt}\gtdefault}
593 \g@addto@macro\@ttfamilyhook
594 {\prepare@family@series@update@kanji{gt}\gtdefault}
595 \fi
596 \else % --- for 2020-02-02 or older END & for 2020-10-01 BEGIN
597 \AddToHook{rmfamily}%
598 {\prepare@family@series@update@kanji{mc}\mcdefault}
599 \AddToHook{sffamily}%
600 {\prepare@family@series@update@kanji{gt}\gtdefault}
601 \AddToHook{ttfamily}%
602 {\prepare@family@series@update@kanji{gt}\gtdefault}
603 \fi % --- for 2020-10-01 END
604 \</class|minijs>
605 \<*class>

```

`\textmc` 次のコマンドはイタリック補正なども含めて定義されていますが、和文ではイタリック補正はあまり役に立たず、欧文・和文間のグルーが入らないという副作用もありますので、単純な定義に直します。

[2016-08-26] 和欧文間の `\xkanjiskip` が入らない問題は、`plfonts.dtx v1.3i` (2000/07/13) の時点で修正されていました。逆に、`amsmath` パッケージを読み込んだ場合に、数式内の添字で文字サイズが変化するようになるはずのところが、変わらなくなっていましたので、修正しました。

[2017-09-03] Yue ZHANG さん作の `fixjfm` パッケージが `\documentclass` より前に `\RequirePackage{fixjfm}` として読み込まれていた場合には、その定義を優先するため、このクラスファイルでは再定義しません。

[2017-09-19] 2010 年の `pTeX` の修正で、イタリック補正と和欧文間の `\xkanjiskip` の衝突が起きなくなっていますから、もうここにあるような単純化は必要ありません。ただし、このクラスファイルが古い `TeX` 環境で利用される可能性も捨てきれないので、とりあえず残しておきます。

```

606 \ifx\DeclareFixJFMCJKTextFontCommand\undefined
607 \DeclareRobustCommand\textmc[1]{%
608   \relax\ifmmode \expandafter\nfss@text \fi{\mcfamily #1}}
609 \DeclareRobustCommand\textgt[1]{%
610   \relax\ifmmode \expandafter\nfss@text \fi{\gtfamily #1}}
611 \fi

```

新クラスでも `disablejfam` オプションを与えなければ数式内で日本語が使えるようにしました。

さらに 2005/12/01 版の `LaTeX` に対応した `pLaTeX` に対応しました (Thanks: ymt さん)。

[2010-03-14] <http://oku.edu.mie-u.ac.jp/tex/mod/forum/discuss.php?d=411> で の山本さんのご指摘に従って修正しました。

```

612 \def\reDeclareMathAlphabet#1#2#3{%
613   \edef\@tempa{\expandafter\@gobble\string#2}%

```

```

614 \edef\@tempb{\expandafter\@gobble\string#3}%
615 \edef\@tempc{\string @\expandafter\@gobbletwo\string#2}%
616 \ifx\@tempc\@tempa%
617   \edef\@tempa{\expandafter\@gobbletwo\string#2}%
618   \edef\@tempb{\expandafter\@gobbletwo\string#3}%
619 \fi
620 \begingroup
621   \let\protect\noexpand
622   \def\@tempaa{\relax}%
623   \expandafter\ifx\csname RDMAorg@\@tempa\endcsname\relax
624     \edef\@tempaa{\expandafter\def\expandafter\noexpand%
625       \csname RDMAorg@\@tempa\endcsname{%
626         \expandafter\noexpand\csname\@tempa\endcsname}}%
627   \fi
628   \def\@tempbb{\relax}%
629   \expandafter\ifx\csname RDMAorg@\@tempb\endcsname\relax
630     \edef\@tempbb{\expandafter\def\expandafter\noexpand%
631       \csname RDMAorg@\@tempb\endcsname{%
632         \expandafter\noexpand\csname\@tempb\endcsname}}%
633   \fi
634   \edef\@tempc{\@tempaa\@tempbb}%
635 \expandafter\endgroup\@tempc%
636 \edef#1{\noexpand\protect\expandafter\noexpand\csname%
637   \expandafter\@gobble\string#1\space\space\endcsname}%
638 \expandafter\edef\csname\expandafter\@gobble\string#1\space\space\endcsname%
639   {\noexpand\DualLang@mathalph@bet%
640     {\expandafter\noexpand\csname RDMAorg@\@tempa\endcsname}%
641     {\expandafter\noexpand\csname RDMAorg@\@tempb\endcsname}%
642   }%
643 }
644 \@onlypreamble\reDeclareMathAlphabet
645 \def\DualLang@mathalph@bet#1#2{%
646   \relax\ifmmode
647     \ifx\math@bgroup\bgroup%      2e normal style      (\mathrm{...})
648       \bgroup\let\DualLang@Mfontsw\DLMfontsw@standard
649     \else
650       \ifx\math@bgroup\relax%      2e two letter style (\rm->\mathrm)
651         \let\DualLang@Mfontsw\DLMfontsw@oldstyle
652       \else
653         \ifx\math@bgroup\@empty% 2.09 oldlfont style ({\mathrm ...})
654           \let\DualLang@Mfontsw\DLMfontsw@oldlfont
655         \else%                      panic! assume 2e normal style
656           \bgroup\let\DualLang@Mfontsw\DLMfontsw@standard
657         \fi
658       \fi
659     \fi
660   \else
661     \let\DualLang@Mfontsw\@firstoftwo
662   \fi

```

```

663 \DualLang@Mfontsw{#1}{#2}%
664 }
665 \def\DLMfontsw@standard#1#2#3{#1{#2{#3}}\egroup}
666 \def\DLMfontsw@oldstyle#1#2{#1\relax\@fontswitch\relax{#2}}
667 \def\DLMfontsw@oldfont#1#2{#1\relax#2\relax}
668 \if@enablejfam
669 \DeclareSymbolFont{mincho}{\jsc@JYn}{mc}{m}{n}
670 \DeclareSymbolFontAlphabet{\mathmc}{mincho}
671 \SetSymbolFont{mincho}{bold}{\jsc@JYn}{gt}{m}{n}
672 \jfam\symmincho
673 \DeclareMathAlphabet{\mathgt}{\jsc@JYn}{gt}{m}{n}
674 \AtBeginDocument{%
675 \reDeclareMathAlphabet{\mathrm}{\@mathrm}{\@mathmc}
676 \reDeclareMathAlphabet{\mathbf}{\@mathbf}{\@mathgt}}
677 \fi

```

`\textsterling` これは `\pounds` 命令で実際に呼び出される文字です。従来からの OT1 エンコーディングでは `\$` のイタリック体が `\pounds` なので `cmti` が使われていましたが、1994 年春からは `cmu` (upright italic, 直立イタリック体) に変わりました。しかし `cmu` はその性格からして実験的なものであり、`\pounds` 以外で使われるとは思えないので、ここでは `cmti` に戻してしまいます。

[2003-08-20] Computer Modern フォントを使う機会も減り、T1 エンコーディングが一般的になってきました。この定義はもうあまり意味がないので消します。

```

678 % \DeclareTextCommand{\textsterling}{OT1}{\itshape\char'\$}

```

禁則パラメータも若干修正します。

アスキーの `kinsoku.dtx` では次の三つが 5000 に設定されています。これを 10000 に再設定します。

```

679 \prebreakpenalty\jis"2147=10000 % 5000 '
680 \postbreakpenalty\jis"2148=10000 % 5000 "
681 \prebreakpenalty\jis"2149=10000 % 5000 "

```

「`TEX` !」「`〒515`」の記号と数字の間に四分アキが入らないようにします。

```

682 \inhibitxspcode'!=1
683 \inhibitxspcode'〒=2

```

以前の版では、たとえば「ベース名. 拡張子」のように和文文字で書いたとき、ピリオドの後に四分アキが入らないようにするために

```

684 % \xspcode' .=0

```

のようにしていました。ただ、「Foo Inc. は……」のように書いたときにもスペースが入らなくなるので、ちょっとまずい修正だったかもしれません。元に戻しました。

とりあえず「ベース名.`\mbox{}`拡張子」と書いてください。

「C や C++ では……」と書くと、C++ の直後に四分アキが入らないのでバランスが悪くなります。四分アキが入るようにしました。% の両側も同じです。

```

685 \xspcode'+=3
686 \xspcode'\%=3

```

これ以外に T1 エンコーディングで 80~ff の文字もすべて欧文文字ですので、両側の和文文字との間にスペースが入らなければなりません。

```
687 \xspcode'^^80=3
688 \xspcode'^^81=3
689 \xspcode'^^82=3
690 \xspcode'^^83=3
691 \xspcode'^^84=3
692 \xspcode'^^85=3
693 \xspcode'^^86=3
694 \xspcode'^^87=3
695 \xspcode'^^88=3
696 \xspcode'^^89=3
697 \xspcode'^^8a=3
698 \xspcode'^^8b=3
699 \xspcode'^^8c=3
700 \xspcode'^^8d=3
701 \xspcode'^^8e=3
702 \xspcode'^^8f=3
703 \xspcode'^^90=3
704 \xspcode'^^91=3
705 \xspcode'^^92=3
706 \xspcode'^^93=3
707 \xspcode'^^94=3
708 \xspcode'^^95=3
709 \xspcode'^^96=3
710 \xspcode'^^97=3
711 \xspcode'^^98=3
712 \xspcode'^^99=3
713 \xspcode'^^9a=3
714 \xspcode'^^9b=3
715 \xspcode'^^9c=3
716 \xspcode'^^9d=3
717 \xspcode'^^9e=3
718 \xspcode'^^9f=3
719 \xspcode'^^a0=3
720 \xspcode'^^a1=3
721 \xspcode'^^a2=3
722 \xspcode'^^a3=3
723 \xspcode'^^a4=3
724 \xspcode'^^a5=3
725 \xspcode'^^a6=3
726 \xspcode'^^a7=3
727 \xspcode'^^a8=3
728 \xspcode'^^a9=3
729 \xspcode'^^aa=3
730 \xspcode'^^ab=3
731 \xspcode'^^ac=3
732 \xspcode'^^ad=3
```

733 \xspcode'^^ae=3  
734 \xspcode'^^af=3  
735 \xspcode'^^b0=3  
736 \xspcode'^^b1=3  
737 \xspcode'^^b2=3  
738 \xspcode'^^b3=3  
739 \xspcode'^^b4=3  
740 \xspcode'^^b5=3  
741 \xspcode'^^b6=3  
742 \xspcode'^^b7=3  
743 \xspcode'^^b8=3  
744 \xspcode'^^b9=3  
745 \xspcode'^^ba=3  
746 \xspcode'^^bb=3  
747 \xspcode'^^bc=3  
748 \xspcode'^^bd=3  
749 \xspcode'^^be=3  
750 \xspcode'^^bf=3  
751 \xspcode'^^c0=3  
752 \xspcode'^^c1=3  
753 \xspcode'^^c2=3  
754 \xspcode'^^c3=3  
755 \xspcode'^^c4=3  
756 \xspcode'^^c5=3  
757 \xspcode'^^c6=3  
758 \xspcode'^^c7=3  
759 \xspcode'^^c8=3  
760 \xspcode'^^c9=3  
761 \xspcode'^^ca=3  
762 \xspcode'^^cb=3  
763 \xspcode'^^cc=3  
764 \xspcode'^^cd=3  
765 \xspcode'^^ce=3  
766 \xspcode'^^cf=3  
767 \xspcode'^^d0=3  
768 \xspcode'^^d1=3  
769 \xspcode'^^d2=3  
770 \xspcode'^^d3=3  
771 \xspcode'^^d4=3  
772 \xspcode'^^d5=3  
773 \xspcode'^^d6=3  
774 \xspcode'^^d7=3  
775 \xspcode'^^d8=3  
776 \xspcode'^^d9=3  
777 \xspcode'^^da=3  
778 \xspcode'^^db=3  
779 \xspcode'^^dc=3  
780 \xspcode'^^dd=3  
781 \xspcode'^^de=3



```

782 \xspcode'^^df=3
783 \xspcode'^^e0=3
784 \xspcode'^^e1=3
785 \xspcode'^^e2=3
786 \xspcode'^^e3=3
787 \xspcode'^^e4=3
788 \xspcode'^^e5=3
789 \xspcode'^^e6=3
790 \xspcode'^^e7=3
791 \xspcode'^^e8=3
792 \xspcode'^^e9=3
793 \xspcode'^^ea=3
794 \xspcode'^^eb=3
795 \xspcode'^^ec=3
796 \xspcode'^^ed=3
797 \xspcode'^^ee=3
798 \xspcode'^^ef=3
799 \xspcode'^^f0=3
800 \xspcode'^^f1=3
801 \xspcode'^^f2=3
802 \xspcode'^^f3=3
803 \xspcode'^^f4=3
804 \xspcode'^^f5=3
805 \xspcode'^^f6=3
806 \xspcode'^^f7=3
807 \xspcode'^^f8=3
808 \xspcode'^^f9=3
809 \xspcode'^^fa=3
810 \xspcode'^^fb=3
811 \xspcode'^^fc=3
812 \xspcode'^^fd=3
813 \xspcode'^^fe=3
814 \xspcode'^^ff=3

```

\@ 欧文といえば、 $\text{\LaTeX}$  の  $\text{\def\@{\spacefactor\@m}}$  という定義 ( $\text{\@m}$  は 1000) では  $\text{I watch TV\@.}$  と書くと  $V$  とピリオドのペアカーニングが効かなくなります。そこで、次のような定義に直し、 $\text{I watch TV.\@}$  と書くことにします。

[2016-07-14] 2015-01-01 の  $\text{\LaTeX}$  で、auxiliary files に書き出されたときにスペースが食われないようにする修正が入りました。これに合わせて  $\{\}$  を補いました。

```

815 \def\@{\spacefactor3000{}}

```

## 4 フォントサイズ

フォントサイズを変える命令 ( $\text{\normalsize}$ ,  $\text{\small}$  など) の実際の挙動の設定は、三つの引数をとる命令  $\text{\@setfontsize}$  を使って、たとえば

```

\@setfontsize{\normalsize}{10}{16}

```

のように行います。これは

`\normalsize` は 10 ポイントのフォントを使い、行送りは 16 ポイントである

という意味です。ただし、処理を速くするため、以下では 10 と同義の L<sup>A</sup>T<sub>E</sub>X の内部命令 `\@xpt` を使っています。この `\@xpt` の類は次のものがあり、L<sup>A</sup>T<sub>E</sub>X 本体で定義されています。

<code>\@vpt</code>	5	<code>\@vipt</code>	6	<code>\@viipt</code>	7
<code>\@viipt</code>	8	<code>\@ixpt</code>	9	<code>\@xpt</code>	10
<code>\@xipt</code>	10.95	<code>\@xiipt</code>	12	<code>\@xivpt</code>	14.4

`\@setfontsize` ここでは `\@setfontsize` の定義を少々変更して、段落の字下げ `\parindent`、和文文字間のスペース `\kanjiskip`、和文・欧文間のスペース `\xkanjiskip` を変更しています。

`\kanjiskip` は pL<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> で 0pt plus .4pt minus .5pt に設定していますが、これはそもそも文字サイズの変更に応じて変わるべきものです。それに、プラスになったりマイナスになったりするの、追い出しと追い込みの混在が生じ、統一性を欠きます。なるべく追い出しになるようにプラスの値だけにしたいところですが、ごくわずかなマイナスは許すことにしました。

`\xkanjiskip` については、四分つまり全角の 1/4 を標準として、追い出すために三分あるいは二分まで延ばすのが一般的ですが、ここでは Times や Palatino のスペースがほぼ四分であることに着目して、これに一致させています。これなら書くときにスペースを空けても空けなくても同じ出力になります。

`\parindent` については、0 (以下) でなければ全角幅 (1zw) に直します。

[2008-02-18] english オプションで `\parindent` を 1em にしました。

```
816 </class>
817 <*class|minijs>
818 %% \@setfontsize with \parindent and \(\x)kanjiskip settings
819 \def\@setfontsize#1#2#3{%
820 <minijs> \@nomath#1%
821 \ifx\protect\@typeset@protect
822 \let\@currsize#1%
823 \fi
824 \fontsize{#2}{#3}\selectfont
825 \ifdim\parindent>\z@
826 <class> \if@english
827 <class> \parindent=1em
828 <class> \else
829 \parindent=1zw
830 <class> \fi
831 \fi
832 \kanjiskip=0zw plus .1zw minus .01zw
833 <class> \ifdim\xkanjiskip>\z@
834 <class> \if@slide \xkanjiskip=0.1em \else
835 \xkanjiskip=0.25em plus 0.15em minus 0.06em
836 <class> \fi
```

```

837 <class> \fi
838 }
839 </class | minijs>
840 <*class>

```

`\jsc@setfontsize` クラスファイルの内部では、拡大率も考慮した `\jsc@setfontsize` を `\@setfontsize` の代わりに用いることにします。

```

841 \def\jsc@setfontsize#1#2#3{%
842   \@setfontsize#1{#2\jsc@mpt}{#3\jsc@mpt}}

```

これらのグルーをもってしても行分割ができない場合は、`\emergencystretch` に訴えます。

```

843 \emergencystretch 3zw

```

`\ifnarrowbaselines` 欧文用に行間を狭くする論理変数と、それを真・偽にするためのコマンドです。

`\narrowbaselines` [2003-06-30] 数式に入るところで `\narrowbaselines` を実行しているので `\abovedisplayskip` 等が初期化されてしまうという shintok さんのご指摘に対して、しっぽ愛好家さんが次の修正を教えてくださいました。

[2008-02-18] `english` オプションで最初の段落のインデントをしないようにしました。

TODO: Hasumi さん [qa:54539] のご指摘は考慮中です。

```

844 \newif\ifnarrowbaselines
845 \if@english
846   \narrowbaselinestrue
847 \fi
848 \def\narrowbaselines{%
849   \narrowbaselinestrue
850   \skip0=\abovedisplayskip
851   \skip2=\abovedisplayshortskip
852   \skip4=\belowdisplayskip
853   \skip6=\belowdisplayshortskip
854   \@currsize\selectfont
855   \abovedisplayskip=\skip0
856   \abovedisplayshortskip=\skip2
857   \belowdisplayskip=\skip4
858   \belowdisplayshortskip=\skip6\relax}
859 \def\widebaselines{\narrowbaselinesfalse\@currsize\selectfont}

```

`\normalsize` 標準のフォントサイズと行送りを選ぶコマンドです。

本文 10 ポイントのときの行送りは、欧文の標準クラスファイルでは 12 ポイント、アスキーの和文クラスファイルでは 15 ポイントになっていますが、ここでは 16 ポイントにしました。ただし `\narrowbaselines` で欧文用の 12 ポイントになります。

公称 10 ポイントの和文フォントが約 9.25 ポイント（アスキーのものの 0.961 倍）であることもあり、行送りがかなりゆったりとしたと思います。実際、 $16/9.25 \approx 1.73$  であり、和文の推奨値の一つ「二分四分」（1.75）に近づきました。

```

860 \renewcommand{\normalsize}{%
861   \ifnarrowbaselines

```

```

862 \jsc@setfontsize\normalsize\@xpt\@xipt
863 \else
864 \jsc@setfontsize\normalsize\@xpt{\n@baseline}%
865 \fi

```

数式の上のアキ(`\abovedisplayskip`), 短い数式の上のアキ(`\abovedisplayshortskip`), 数式の下のアキ (`\belowdisplayshortskip`) の設定です。

[2003-02-16] ちょっと変えました。

[2009-08-26] T<sub>E</sub>X Q & A 52569 から始まる議論について逡巡していましたが, 結局, 微調節してみることにしました。

```

866 \abovedisplayskip 11\jsc@empt \@plus3\jsc@empt \@minus4\jsc@empt
867 \abovedisplayshortskip \z@ \@plus3\jsc@empt
868 \belowdisplayskip 9\jsc@empt \@plus3\jsc@empt \@minus4\jsc@empt
869 \belowdisplayshortskip \belowdisplayskip

```

最後に, リスト環境のトップレベルのパラメータ `\@listI` を, `\@listi` にコピーしておきます。`\@listI` の設定は後で出てきます。

```

870 \let\@listi\@listI}

```

ここで実際に標準フォントサイズで初期化します。

```

871 </class>
872 <*class | minijs>
873 %% initialize
874 \normalsize
875 </class | minijs>
876 <*class>

```

`\Cht` 基準となる長さの設定をします。pL<sub>A</sub>T<sub>E</sub>X 2<sub>ε</sub> カーネル (`plfonts.dtx`) で宣言されているパ

`\Cdp` ラメータに実際の値を設定します。たとえば `\Cwd` は `\normalfont` の全角幅 (1zw) です。

`\Cwd` [2017-08-31] 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` [2017-09-19] 内部的に使った `\box0` を空にします。

```

877 \setbox0\hbox{\char\jis"3441}%
878 \setlength\Cht{\ht0}
879 \setlength\Cdp{\dp0}
880 \setlength\Cwd{\wd0}
881 \setlength\Cvs{\baselineskip}
882 \setlength\Chs{\wd0}
883 \setbox0=\box\voidb@x

```

`\small` `\small` も `\normalsize` と同様に設定します。行送りは, `\normalsize` が 16 ポイントなら, 割合からすれば  $16 \times 0.9 = 14.4$  ポイントになりますが, `\small` の使われ方を考えて, ここでは和文 13 ポイント, 欧文 11 ポイントとします。また, `\topsep` と `\parsep` は, 元はそれぞれ  $4 \pm 2$ ,  $2 \pm 1$  ポイントでしたが, ここではゼロ (`\z@`) にしました。

```

884 \newcommand{\small}{%
885 \ifnarrowbaselines
886 \!kiyou) \jsc@setfontsize\small\@ixpt{11}%

```

```

887 <kiyou> \jsc@setfontsize\small{8.8888}{11}%
888 \else
889 <!kiyou> \jsc@setfontsize\small\@ixpt{13}%
890 <kiyou> \jsc@setfontsize\small{8.8888}{13.2418}%
891 \fi
892 \abovedisplayskip 9\jsc@mpt \@plus3\jsc@mpt \@minus4\jsc@mpt
893 \abovedisplayshortskip \z@ \@plus3\jsc@mpt
894 \belowdisplayskip \abovedisplayskip
895 \belowdisplayshortskip \belowdisplayskip
896 \def\@listi{\leftmargin\leftmargin
897 \topsep \z@
898 \parsep \z@
899 \itemsep \parsep}}

```

\footnotesize \footnotesize も同様です。 \topsep と \parsep は、元はそれぞれ  $3 \pm 1$ 、 $2 \pm 1$  ポイントでしたが、ここではゼロ (\z@) にしました。

```

900 \newcommand{\footnotesize}{%
901 \ifnarrowbaselines
902 <!kiyou> \jsc@setfontsize\footnotesize\@viipt{9.5}%
903 <kiyou> \jsc@setfontsize\footnotesize{8.8888}{11}%
904 \else
905 <!kiyou> \jsc@setfontsize\footnotesize\@viipt{11}%
906 <kiyou> \jsc@setfontsize\footnotesize{8.8888}{13.2418}%
907 \fi
908 \abovedisplayskip 6\jsc@mpt \@plus2\jsc@mpt \@minus3\jsc@mpt
909 \abovedisplayshortskip \z@ \@plus2\jsc@mpt
910 \belowdisplayskip \abovedisplayskip
911 \belowdisplayshortskip \belowdisplayskip
912 \def\@listi{\leftmargin\leftmargin
913 \topsep \z@
914 \parsep \z@
915 \itemsep \parsep}}

```

\scriptsize それ以外のサイズは、本文に使うことがないので、単にフォントサイズと行送りだけ変更します。特に注意すべきは \large で、これは二段組のときに節見出しのフォントとして使い、行送りを \normalsize と同じにすることによって、節見出しが複数行にわたっても段間で行が揃うようにします。

\LARGE [2004-11-03] \HUGE を追加。

```

\huge 916 \newcommand{\scriptsize}{\jsc@setfontsize\scriptsize\@viipt\@viipt}
917 \newcommand{\tiny}{\jsc@setfontsize\tiny\@vpt\@vpt}
\LARGE 918 \if@twocolumn
\HUGE 919 <!kiyou> \newcommand{\large}{\jsc@setfontsize\large\@xipt{\n@baseline}}
920 <kiyou> \newcommand{\large}{\jsc@setfontsize\large{11.111}{\n@baseline}}
921 \else
922 <!kiyou> \newcommand{\large}{\jsc@setfontsize\large\@xipt{17}}
923 <kiyou> \newcommand{\large}{\jsc@setfontsize\large{11.111}{17}}
924 \fi
925 <!kiyou> \newcommand{\Large}{\jsc@setfontsize\Large\@xivpt{21}}

```

```

926 \newcommand{\Large}{\jsc@setfontsize\Large{12.222}{21}}
927 \newcommand{\LARGE}{\jsc@setfontsize\LARGE{@xvipt{25}}
928 \newcommand{\huge}{\jsc@setfontsize\huge{@xxpt{28}}
929 \newcommand{\Huge}{\jsc@setfontsize\Huge{@xxvpt{33}}
930 \newcommand{\HUGE}{\jsc@setfontsize\HUGE{30}{40}}

```

別行立て数式の中では `\narrowbaselines` にします。和文の行送りのままでは、行列や場合分けの行送り、連分数の高さなどが不釣り合いに大きくなるためです。

本文中の数式の中では `\narrowbaselines` にしていません。本文中ではなるべく行送りが変わるような大きいものを使わず、行列は `amsmath` の `smallmatrix` 環境を使うのがいいでしょう。

```

931 \everydisplay=\expandafter{\the\everydisplay \narrowbaselines}

```

しかし、このおかげで別行数式の上下のスペースが少し違ってしまいました。とりあえず `amsmath` の `equation` 関係は `okumacro` のほうで逃げていますが、もっとうまい逃げ道があれば教えてください。

見出し用のフォントは `\bfseries` 固定ではなく、`\headfont` という命令で定めることにします。これは太ゴシックが使えるときは `\sffamily \bfseries` でいいと思いますが、通常の中ゴシックでは単に `\sffamily` だけのほうがよさそうです。『*LaTeX 2<sub>ε</sub>* 美文書作成入門』(1997 年)では `\sffamily \fontseries{sbc}` として新ゴ M と合わせましたが、`\fontseries{sbc}` はちょっと幅が狭いように感じました。

```

932 % \newcommand{\headfont}{\bfseries}
933 \newcommand{\headfont}{\gtfamily\sffamily}
934 % \newcommand{\headfont}{\sffamily\fontseries{sbc}\selectfont}

```

## 5 レイアウト

### ■二段組

`\columnsep` `\columnsep` は二段組のときの左右の段間の幅です。元は 10pt ですが、2zw にしました。  
`\columnseprule` このスペースの中央に `\columnseprule` の幅の罫線が引かれます。

```

935 \setlength\columnsep{2zw}
936 \setlength\columnsep{28truebp}
937 \setlength\columnseprule{\z@}

```

### ■段落

`\lineskip` 上下の行の文字が `\lineskiplimit` より接近したら、`\lineskip` より近づかないようにします。元は 0pt ですが 1pt に変更しました。`normal...` の付いた方は保存用です。  
`\normallineskip`  
`\lineskiplimit` 938 \setlength\lineskip{1\jsc@empt}  
`\normallineskiplimit` 939 \setlength\normallineskip{1\jsc@empt}  
940 \setlength\lineskiplimit{1\jsc@empt}  
941 \setlength\normallineskiplimit{1\jsc@empt}  
`\baselinestretch` 実際の行送りが `\baselineskip` の何倍かを表すマクロです。たとえば

```
\renewcommand{\baselinestretch}{2}
```

とすると、行送りが通常の 2 倍になります。ただし、これを設定すると、たとえ `\baselineskip` が伸縮するように設定しても、行送りの伸縮ができなくなります。行送りの伸縮はしないのが一般的です。

```
942 \renewcommand{\baselinestretch}{}
```

`\parskip` `\parskip` は段落間の追加スペースです。元は 0pt plus 1pt になっていましたが、ここでは `\parindent` ゼロにしました。`\parindent` は段落の先頭の字下げ幅です。

```
943 \setlength\parskip{\z@}
944 \if@slide
945   \setlength\parindent{0zw}
946 \else
947   \setlength\parindent{1zw}
948 \fi
```

`\@lowpenalty` `\nopagebreak`, `\nolinebreak` は引数に応じて次のペナルティ値のうちどれかを選ぶようになっています。ここはオリジナル通りです。

```
\@highpenalty 949 \@lowpenalty 51
950 \@medpenalty 151
951 \@highpenalty 301
```

`\interlinepenalty` 段落中の改ページのペナルティです。デフォルトは 0 です。

```
952 % \interlinepenalty 0
```

`\brokenpenalty` ページの最後の行がハイフンで終わる際のペナルティです。デフォルトは 100 です。

```
953 % \brokenpenalty 100
```

## 5.1 ページレイアウト

### ■縦方向のスペース

`\headheight` `\topskip` は本文領域上端と本文 1 行目のベースラインとの距離です。あまりぎりぎりの値にすると、本文中に  $\int$  のような高い文字が入ったときに 1 行目のベースラインが他のページより下がってしまいます。ここでは本文の公称フォントサイズ (10pt) にします。

[2003-06-26] `\headheight` はヘッダの高さで、元は 12pt でしたが、新ドキュメントクラスでは `\topskip` と等しくしていました。ところが、`fancyhdr` パッケージで `\headheight` が小さいとおかしいことになるようですので、2 倍に増やしました。代わりに、版面の上下揃えの計算では `\headheight` ではなく `\topskip` を使うことにしました。

[2016-08-17] 圏点やルビが一行目に来た場合に下がるのを防ぐため、`\topskip` を 10pt から 1.38zw に増やしました。`\headheight` は従来と同じ 20pt のままとします。

```
954 \setlength\topskip{1.38zw}%% from 10\jsc@empt (2016-08-17)
955 \if@slide
956   \setlength\headheight{0\jsc@empt}
957 \else
```

```

958 \setlength\headheight{20\jsc@empt}%% from 2\topskip (2016-08-17); from \topskip (2003-06-26)
959 \fi

```

`\footskip` `\footskip` は本文領域下端とフッタ下端との距離です。標準クラスファイルでは、book で 0.35in (約 8.89mm), book 以外で 30pt (約 10.54mm) となっていました。ここでは A4 判のときちょうど 1cm となるように、`\paperheight` の 0.03367 倍 (最小 `\baselineskip`) としました。書籍については、フッタは使わないことにして、ゼロにしました。

```

960 < *article | kiyou >
961 \if@slide
962 \setlength\footskip{0pt}
963 \else
964 \setlength\footskip{0.03367\paperheight}
965 \ifdim\footskip<\baselineskip
966 \setlength\footskip{\baselineskip}
967 \fi
968 \fi
969 < /article | kiyou >
970 \jpf\setlength\footskip{9\jsc@mmm}
971 < *book >
972 \if@report
973 \setlength\footskip{0.03367\paperheight}
974 \ifdim\footskip<\baselineskip
975 \setlength\footskip{\baselineskip}
976 \fi
977 \else
978 \setlength\footskip{0pt}
979 \fi
980 < /book >
981 < *report >
982 \setlength\footskip{0.03367\paperheight}
983 \ifdim\footskip<\baselineskip
984 \setlength\footskip{\baselineskip}
985 \fi
986 < /report >

```

`\headsep` `\headsep` はヘッダ下端と本文領域上端との距離です。元は book で 18pt (約 6.33mm), それ以外で 25pt (約 8.79mm) になっていました。ここでは article は `\footskip - \topskip` としました。

[2016-10-08] article の slide のとき、および book の非 report と kiyou のときに `\headsep` を減らしそこねていたのを修正しました (2016-08-17 での修正漏れ)。

```

987 < *article >
988 \if@slide
989 \setlength\headsep{0\jsc@empt}
990 \addtolength\headsep{-\topskip}%% added (2016-10-08)
991 \addtolength\headsep{10\jsc@empt}%% added (2016-10-08)
992 \else
993 \setlength\headsep{\footskip}

```



```

994 \addtolength\headsep{-\topskip}
995 \fi
996 </article>
997 <*book>
998 \if@report
999 \setlength\headsep{\footskip}
1000 \addtolength\headsep{-\topskip}
1001 \else
1002 \setlength\headsep{6\jsc@mmm}
1003 \addtolength\headsep{-\topskip}%% added (2016-10-08)
1004 \addtolength\headsep{10\jsc@mpt}%% added (2016-10-08)
1005 \fi
1006 </book>
1007 <*report>
1008 \setlength\headsep{\footskip}
1009 \addtolength\headsep{-\topskip}
1010 </report>
1011 <*jspf>
1012 \setlength\headsep{9\jsc@mmm}
1013 \addtolength\headsep{-\topskip}
1014 </jspf>
1015 <*kiyou>
1016 \setlength\headheight{0\jsc@mpt}
1017 \setlength\headsep{0\jsc@mpt}
1018 \addtolength\headsep{-\topskip}%% added (2016-10-08)
1019 \addtolength\headsep{10\jsc@mpt}%% added (2016-10-08)
1020 </kiyou>

```

`\maxdepth` `\maxdepth` は本文最下行の最大の深さで、plain  $\mathrm{T\!E\!X}$  や  $\mathrm{L\!A\!T\!E\!X}$  2.09 では 4pt に固定でした。 $\mathrm{L\!A\!T\!E\!X}$  2e では `\maxdepth + \topskip` を本文フォントサイズの 1.5 倍にしたいのですが、`\topskip` は本文フォントサイズ（ここでは 10pt）に等しいので、結局 `\maxdepth` は `\topskip` の半分の値（具体的には 5pt）にします。

```

1021 \setlength\maxdepth{.5\topskip}

```

### ■本文の幅と高さ

`\fullwidth` 本文の幅が全角 40 文字を超えると読みにくくなります。そこで、書籍の場合に限って、紙の幅が広いときは外側のマージンを余分にとって全角 40 文字に押え、ヘッダやフッタは本文領域より広く取ることにします。このときヘッダやフッタの幅を表す `\fullwidth` という長さを定義します。

```

1022 \newdimen\fullwidth

```

この `\fullwidth` は article では紙幅 `\paperwidth` の 0.76 倍を超えない全角幅の整数倍（二段組では全角幅の偶数倍）にします。0.76 倍という数値は A4 縦置きの場合に紙幅から約 2 インチを引いた値になるように選びました。book では紙幅から 36 ミリを引いた値にしました。

`\textwidth` 書籍以外では本文領域の幅 `\textwidth` は `\fullwidth` と等しくします。article では

A4 縦置きで 49 文字となります。某学会誌スタイルでは 50zw (25 文字×2 段) + 段間 8mm とします。

```

1023 <*article>
1024 \if@slide
1025   \setlength\fullwidth{0.9\paperwidth}
1026 \else
1027   \setlength\fullwidth{0.76\paperwidth}
1028 \fi
1029 \if@twocolumn \@tempdima=2zw \else \@tempdima=1zw \fi
1030 \divide\fullwidth\@tempdima \multiply\fullwidth\@tempdima
1031 \setlength\textwidth{\fullwidth}
1032 </article>
1033 <*book>
1034 \if@report
1035   \setlength\fullwidth{0.76\paperwidth}
1036 \else
1037   \setlength\fullwidth{\paperwidth}
1038   \addtolength\fullwidth{-36\jsc@mmm}
1039 \fi
1040 \if@twocolumn \@tempdima=2zw \else \@tempdima=1zw \fi
1041 \divide\fullwidth\@tempdima \multiply\fullwidth\@tempdima
1042 \setlength\textwidth{\fullwidth}
1043 \if@report \else
1044   \if@twocolumn \else
1045     \ifdim \fullwidth>40zw
1046       \setlength\textwidth{40zw}
1047     \fi
1048   \fi
1049 \fi
1050 </book>
1051 <*report>
1052 \setlength\fullwidth{0.76\paperwidth}
1053 \if@twocolumn \@tempdima=2zw \else \@tempdima=1zw \fi
1054 \divide\fullwidth\@tempdima \multiply\fullwidth\@tempdima
1055 \setlength\textwidth{\fullwidth}
1056 </report>
1057 <*jspf>
1058 \setlength\fullwidth{50zw}
1059 \addtolength\fullwidth{8\jsc@mmm}
1060 \setlength\textwidth{\fullwidth}
1061 </jspf>
1062 <*kiyou>
1063 \setlength\fullwidth{48zw}
1064 \addtolength\fullwidth{\columnsep}
1065 \setlength\textwidth{\fullwidth}
1066 </kiyou>

```

\textheight 紙の高さ \paperheight は、1 インチと \topmargin と \headheight と \headsep と

`\textheight` と `\footskip` とページ下部の余白を加えたものです。

本文部分の高さ `\textheight` は、紙の高さ `\paperheight` の 0.83 倍から、ヘッダの高さ、ヘッダと本文の距離、本文とフッタ下端の距離、`\topskip` を引き、それを `\baselineskip` の倍数に切り捨て、最後に `\topskip` を加えます。念のため 0.1 ポイント余分に加えておきます。0.83 倍という数値は、A4 縦置きの場合に紙の高さから上下マージン各約 1 インチを引いた値になるように選びました。

某学会誌スタイルでは 44 行にします。

[2003-06-26] `\headheight` を `\topskip` に直しました。以前はこの二つは値が同じであったので、変化はないはずです。

[2016-08-26] `\topskip` を 10pt から 1.38zw に増やしましたので、その分 `\textheight` を増やします (2016-08-17 での修正漏れ)。

[2016-10-08] article の slide のときに `\headheight` はゼロなので、さらに修正しました (2016-08-17 での修正漏れ)。

```
1067 (*article | book | report)
1068 \if@slide
1069   \setlength{\textheight}{0.95\paperheight}
1070 \else
1071   \setlength{\textheight}{0.83\paperheight}
1072 \fi
1073 \addtolength{\textheight}{-10\jsc@mpt}%% from -\topskip (2016-10-08); from -\headheight (2003-
1074 \addtolength{\textheight}{-\headsep}
1075 \addtolength{\textheight}{-\footskip}
1076 \addtolength{\textheight}{-\topskip}
1077 \divide\textheight\baselineskip
1078 \multiply\textheight\baselineskip
1079 </article | book | report>
1080 \jpf\setlength{\textheight}{51\baselineskip}
1081 \kiyou\setlength{\textheight}{47\baselineskip}
1082 \addtolength{\textheight}{\topskip}
1083 \addtolength{\textheight}{0.1\jsc@mpt}
1084 \jpf\setlength{\mathindent}{10\jsc@mm}
```

`\flushbottom` [2016-07-18] `\textheight` に念のため 0.1 ポイント余裕を持たせているのと同様に、`\flushbottom` にも余裕を持たせます。元の  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$  での完全な `\flushbottom` の定義は

```
\def\flushbottom{%
  \let\@textbottom\relax \let\@texttop\relax}
```

ですが、次のようにします。

```
1085 \def\flushbottom{%
1086   \def\@textbottom{\vskip \z@ \@plus.1\jsc@mpt}%
1087   \let\@texttop\relax}
```

`\marginparsep` `\marginparsep` は欄外の書き込みと本文との間隔です。`\marginparpush` は欄外の書き込み `\marginparpush` みどろしの最小の間隔です。

```

1088 \setlength\marginparsep{\columnsep}
1089 \setlength\marginparpush{\baselineskip}

```

`\oddsidemargin` それぞれ奇数ページ、偶数ページの左マージンから 1 インチ引いた値です。片面印刷では `\evensidemargin` `\oddsidemargin` が使われます。TEX は上・左マージンに `1truein` を挿入しますが、トンボ関係のオプションが指定されると pLATEX 2<sub>ε</sub> (`plcore.ltx`) はトンボの内側に `1in` のスペース (`1truein` ではなく) を挿入するので、場合分けしています。

```

1090 \setlength{\oddsidemargin}{\paperwidth}
1091 \addtolength{\oddsidemargin}{-\fullwidth}
1092 \setlength{\oddsidemargin}{.5\oddsidemargin}
1093 \iftombow
1094   \addtolength{\oddsidemargin}{-1in}
1095 \else
1096   \addtolength{\oddsidemargin}{-\inv@mag in}
1097 \fi
1098 \setlength{\evensidemargin}{\oddsidemargin}
1099 \if@mparswitch
1100   \addtolength{\evensidemargin}{\fullwidth}
1101   \addtolength{\evensidemargin}{-\textwidth}
1102 \fi

```

`\marginparwidth` `\marginparwidth` は欄外の書き込みの横幅です。外側マージンの幅 (`\evensidemargin` + 1 インチ) から 1 センチを引き、さらに `\marginparsep` (欄外の書き込みと本文のアキ) を引いた値にしました。最後に `1zw` の整数倍に切り捨てます。

```

1103 \setlength\marginparwidth{\paperwidth}
1104 \addtolength\marginparwidth{-\oddsidemargin}
1105 \addtolength\marginparwidth{-\inv@mag in}
1106 \addtolength\marginparwidth{-\textwidth}
1107 \addtolength\marginparwidth{-10\jsc@mmm}
1108 \addtolength\marginparwidth{-\marginparsep}
1109 \@tempdima=1zw
1110 \divide\marginparwidth\@tempdima
1111 \multiply\marginparwidth\@tempdima

```

`\topmargin` 上マージン (紙の上端とヘッダ上端の距離) から 1 インチ引いた値です。

[2003-06-26] `\headheight` を `\topskip` に直しました。以前はこの二つは値が同じであったので、変化はないはずです。

[2016-08-17] `\topskip` を 10pt から 1.38zw に直しましたが、`\topmargin` は従来の値から変わらないように調節しました。…のつもりでしたが、`\textheight` を増やし忘れていたので変わってしまっていました (2016-08-26 修正済み)。

```

1112 \setlength\topmargin{\paperheight}
1113 \addtolength\topmargin{-\textheight}
1114 \if@slide
1115   \addtolength\topmargin{-\headheight}
1116 \else
1117   \addtolength\topmargin{-10\jsc@empt}%% from -\topskip (2016-10-08); from -\headheight (2003-0
1118 \fi

```

```

1119 \addtolength\topmargin{-\headsep}
1120 \addtolength\topmargin{-\footskip}
1121 \setlength\topmargin{0.5\topmargin}
1122 (kiyou)\setlength\topmargin{81truebp}
1123 \iftombow
1124 \addtolength\topmargin{-1in}
1125 \else
1126 \addtolength\topmargin{-\inv@mag in}
1127 \fi

```

## ■脚注

`\footnotesep` 各脚注の頭に入る支柱 (strut) の高さです。脚注間に余分のアキが入らないように、`\footnotesize` の支柱の高さ (行送りの 0.7 倍) に等しくします。

```

1128 {\footnotesize\global\setlength\footnotesep{\baselineskip}}
1129 \setlength\footnotesep{0.7\footnotesep}

```

`\footins` `\skip\footins` は本文の最終行と最初の脚注との間の距離です。標準の 10 ポイントクラスでは 9 plus 4 minus 2 ポイントになっていますが、和文の行送りを考えてもうちょっと大きくします。

```

1130 \setlength{\skip\footins}{16\jsc@mp\ @plus 5\jsc@mp\ @minus 2\jsc@mp\}

```

■フロート関連 フロート (図, 表) 関連のパラメータは L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> 本体で定義されていますが、ここで設定変更します。本文ページ (本文とフロートが共存するページ) とフロートだけのページで設定が異なります。ちなみに、カウンタは内部では `\c@` を名前に冠したマクロになっています。

`\c@topnumber` `topnumber` カウンタは本文ページ上部のフロートの最大数です。  
[2003-08-23] ちょっと増やしました。

```

1131 \setcounter{topnumber}{9}

```

`\topfraction` 本文ページ上部のフロートが占有できる最大の割合です。フロートが入りやすいように、元の値 0.7 を 0.8 [2003-08-23: 0.85] に変えてあります。

```

1132 \renewcommand{\topfraction}{.85}

```

`\c@bottomnumber` `bottomnumber` カウンタは本文ページ下部のフロートの最大数です。  
[2003-08-23] ちょっと増やしました。

```

1133 \setcounter{bottomnumber}{9}

```

`\bottomfraction` 本文ページ下部のフロートが占有できる最大の割合です。元は 0.3 でした。

```

1134 \renewcommand{\bottomfraction}{.8}

```

`\c@totalnumber` `totalnumber` カウンタは本文ページに入りうるフロートの最大数です。  
[2003-08-23] ちょっと増やしました。

```

1135 \setcounter{totalnumber}{20}

```

`\textfraction` 本文ページに最低限入らなければならない本文の割合です。フロートが入りやすいように元の 0.2 を 0.1 に変えました。

1136 `\renewcommand{\textfraction}{.1}`

`\floatpagefraction` フロートだけのページでのフロートの最小割合です。これも 0.5 を 0.8 に変えてあります。

1137 `\renewcommand{\floatpagefraction}{.8}`

`\c@dbltopnumber` 二段組のとき本文ページ上部に出力できる段抜きフロートの最大数です。

[2003-08-23] ちょっと増やしました。

1138 `\setcounter{dbltopnumber}{9}`

`\dbltopfraction` 二段組のとき本文ページ上部に出力できる段抜きフロートが占めうる最大の割合です。0.7 を 0.8 に変えてあります。

1139 `\renewcommand{\dbltopfraction}{.8}`

`\dblfloatpagefraction` 二段組のときフロートだけのページに入るべき段抜きフロートの最小割合です。0.5 を 0.8 に変えてあります。

1140 `\renewcommand{\dblfloatpagefraction}{.8}`

`\floatsep` `\floatsep` はページ上部・下部のフロート間の距離です。`\textfloatsep` はページ上部・

`\textfloatsep` 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

`\intextsep` 文との距離です。

1141 `\setlength\floatsep {12\jsc@empt \@plus 2\jsc@empt \@minus 2\jsc@empt}`

1142 `\setlength\textfloatsep{20\jsc@empt \@plus 2\jsc@empt \@minus 4\jsc@empt}`

1143 `\setlength\intextsep {12\jsc@empt \@plus 2\jsc@empt \@minus 2\jsc@empt}`

`\dblfloatsep` 二段組のときの段抜きのフロートについての値です。

`\dbltextfloatsep` 1144 `\setlength\dblfloatsep {12\jsc@empt \@plus 2\jsc@empt \@minus 2\jsc@empt}`

1145 `\setlength\dbltextfloatsep{20\jsc@empt \@plus 2\jsc@empt \@minus 4\jsc@empt}`

`\@fptop` フロートだけのページに入るグルーです。`\@fptop` はページ上部, `\@fpbot` はページ下部,

`\@fpsep` `\@fpsep` はフロート間に入ります。

`\@fpbot` 1146 `\setlength\@fptop{0\jsc@empt \@plus 1fil}`

1147 `\setlength\@fpsep{8\jsc@empt \@plus 2fil}`

1148 `\setlength\@fpbot{0\jsc@empt \@plus 1fil}`

`\@dblfpsep` 段抜きフロートについての値です。

`\@dblfpsep` 1149 `\setlength\@dblfpsep{0\jsc@empt \@plus 1fil}`

`\@dblfpbot` 1150 `\setlength\@dblfpsep{8\jsc@empt \@plus 2fil}`

1151 `\setlength\@dblfpbot{0\jsc@empt \@plus 1fil}`

## 6 改ページ（日本語 T<sub>E</sub>X 開発コミュニティ版のみ）

`\pltx@cleartorightpage` [2017-02-24] コミュニティ版 p<sub>A</sub>T<sub>E</sub>X の標準クラス 2017/02/15 に合わせて、同じ命令を追加しました。

`\pltx@cleartooddpage`

`\pltx@cleartoevenpage`

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

となっています。

```

1152 (*article | book | report)
1153 \def\pltx@cleartorightpage{\clearpage\if@twoside
1154   \ifodd\c@page
1155     \iftdir
1156       \hbox{}\thispagestyle{empty}\newpage
1157       \if@twocolumn\hbox{}\newpage\fi
1158     \fi
1159   \else
1160     \ifydir
1161       \hbox{}\thispagestyle{empty}\newpage
1162       \if@twocolumn\hbox{}\newpage\fi
1163     \fi
1164   \fi\fi}
1165 \def\pltx@cleartoleftpage{\clearpage\if@twoside
1166   \ifodd\c@page
1167     \ifydir
1168       \hbox{}\thispagestyle{empty}\newpage
1169       \if@twocolumn\hbox{}\newpage\fi
1170     \fi
1171   \else
1172     \iftdir
1173       \hbox{}\thispagestyle{empty}\newpage
1174       \if@twocolumn\hbox{}\newpage\fi
1175     \fi
1176   \fi\fi}
1177 \def\pltx@cleartooddpage{\clearpage\if@twoside
1178   \ifodd\c@page\else
1179     \hbox{}\thispagestyle{empty}\newpage
1180     \if@twocolumn\hbox{}\newpage\fi
1181   \fi\fi}
1182 \def\pltx@cleartoevenpage{\clearpage\if@twoside
1183   \ifodd\c@page
1184     \hbox{}\thispagestyle{empty}\newpage
1185     \if@twocolumn\hbox{}\newpage\fi
1186   \fi\fi}
1187 (/article | book | report)

```

`\cleardoublepage` [2017-02-24] コミュニティ版 p<sup>A</sup>T<sub>E</sub>X の標準クラス 2017/02/15 に合わせて, report と book クラスの場合に `\cleardoublepage` を再定義します。

```

1188 (*book | report)
1189 \if@openleft
1190   \let\cleardoublepage\pltx@cleartoleftpage

```

```

1191 \else\if@openright
1192 \let\cleardoublepage\pltx@cleartorightpage
1193 \fi\fi
1194 </book|report>

```

## 7 ページスタイル

ページスタイルとして、 $\text{\LaTeX 2}_{\epsilon}$  (欧文版) の標準クラスでは `empty`, `plain`, `headings`, `myheadings` があります。このうち `empty`, `plain` スタイルは  $\text{\LaTeX 2}_{\epsilon}$  本体で定義されています。

アスキーのクラスファイルでは `headnombre`, `footnombre`, `bothstyle`, `jpl@in` が追加されていますが、ここでは欧文標準のものだけにしました。

ページスタイルは `\ps@...` の形のマクロで定義されています。

`\@evenhead` `\@oddhead`, `\@oddfoot`, `\@evenhead`, `\@evenfoot` は偶数・奇数ページの柱 (ヘッダ, フッタ) を出力する命令です。これらは `\fullwidth` 幅の `\hbox` の中で呼び出されます。  
`\@evenfoot` `\ps@...` の中で定義しておきます。

`\@oddfoot` 柱の内容は、`\chapter` が呼び出す `\chaptermark{何々}`, `\section` が呼び出す `\sectionmark{何々}` で設定します。柱を扱う命令には次のものがあります。

```

\markboth{左}{右} 両方の柱を設定します。
\markright{右}     右の柱を設定します。
\leftmark          左の柱を出力します。
\rightmark         右の柱を出力します。

```

柱を設定する命令は、右の柱が左の柱の下位にある場合は十分まともに動作します。たとえば左マークを `\chapter`, 右マークを `\section` で変更する場合はこれにあたります。しかし、同一ページに複数の `\markboth` があると、おかしい結果になることがあります。

`\tableofcontents` のような命令で使われる `\@mkboth` は、`\ps@...` コマンド中で `\markboth` か `\@gobbletwo` (何もしない) に `\let` されます。

`\ps@empty` `empty` ページスタイルの定義です。 $\text{\LaTeX}$  本体で定義されているものをコメントアウトした形で載せておきます。

```

1195 % \def\ps@empty{%
1196 %   \let\@mkboth\@gobbletwo
1197 %   \let\@oddhead\@empty
1198 %   \let\@oddfoot\@empty
1199 %   \let\@evenhead\@empty
1200 %   \let\@evenfoot\@empty}

```

`\ps@plainhead` `plainhead` はシンプルなヘッダだけのページスタイルです。

`\ps@plainfoot` `plainfoot` はシンプルなフッタだけのページスタイルです。

`\ps@plain` `plain` は `book` では `plainhead`, それ以外では `plainfoot` になります。



```

1201 \def\ps@plainfoot{%
1202   \let\@mkboth\@gobbletwo
1203   \let\@oddhead\@empty
1204   \def\@oddfoot{\normalfont\hfil\thepage\hfil}%
1205   \let\@evenhead\@empty
1206   \let\@evenfoot\@oddfoot}
1207 \def\ps@plainhead{%
1208   \let\@mkboth\@gobbletwo
1209   \let\@oddfoot\@empty
1210   \let\@evenfoot\@empty
1211   \def\@evenhead{%
1212     \ifmparswitch \hss \fi
1213     \hbox to \fullwidth{\textbf{\thepage}\hfil}%
1214     \ifmparswitch\else \hss \fi}%
1215   \def\@oddhead{%
1216     \hbox to \fullwidth{\hfil\textbf{\thepage}}\hss}}
1217 \book\if@report \let\ps@plain\ps@plainfoot \else \let\ps@plain\ps@plainhead \fi
1218 \!book\let\ps@plain\ps@plainfoot

```

\ps@headings 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。ここではヘッダにアンダーラインを引くようにしてみました。

まず article の場合です。

```

1219 \*article|kiyou)
1220 \if@twoside
1221   \def\ps@headings{%
1222     \let\@oddfoot\@empty
1223     \let\@evenfoot\@empty
1224     \def\@evenhead{\ifmparswitch \hss \fi
1225       \underline{\hbox to \fullwidth{\textbf{\thepage}\hfil\leftmark}}}%
1226       \ifmparswitch\else \hss \fi}%
1227     \def\@oddhead{%
1228       \underline{%
1229         \hbox to \fullwidth{{\rightmark}\hfil\textbf{\thepage}}}\hss}%
1230     \let\@mkboth\markboth
1231     \def\sectionmark##1{\markboth{%
1232       \ifnum \c@secnumdepth >\z@ \thesection \hskip1zw\fi
1233       ##1}{}}%
1234     \def\subsectionmark##1{\markright{%
1235       \ifnum \c@secnumdepth >\@ne \thesubsection \hskip1zw\fi
1236       ##1}}%
1237   }
1238 \else % if not twoside
1239   \def\ps@headings{%
1240     \let\@oddfoot\@empty
1241     \def\@oddhead{%
1242       \underline{%
1243         \hbox to \fullwidth{{\rightmark}\hfil\textbf{\thepage}}}\hss}%
1244     \let\@mkboth\markboth

```

```

1245 \def\sectionmark##1{\markright{%
1246 \ifnum \c@secnumdepth >\z@ \thesection \hskip1zw\fi
1247 ##1}}
1248 \fi
1249 \end{article} \kiyou

```

次は book および report の場合です。[2011-05-10] しっぱ愛好家さん [qa:6370] のパッチを取り込ませていただきました（北見さん [qa:55896] のご指摘ありがとうございます）。

```

1250 \if*book|report
1251 \newif\if@omit@number
1252 \def\ps@headings{%
1253 \let\@oddfoot\@empty
1254 \let\@evenfoot\@empty
1255 \def\@evenhead{%
1256 \if@mparswitch \hss \fi
1257 \underline{\hbox to \fullwidth{\autoxspacing
1258 \textbf{\thepage}\hfil\leftmark}}}%
1259 \if@mparswitch\else \hss \fi}%
1260 \def\@oddhead{\underline{\hbox to \fullwidth{\autoxspacing
1261 {\if@twoside\rightmark\else\leftmark\fi}\hfil\textbf{\thepage}}}\hss}%
1262 \let\@mkboth\markboth
1263 \def\chaptermark##1{\markboth{%
1264 \ifnum \c@secnumdepth >\m@ne
1265 (book) \if@mainmatter
1266 \if@omit@number\else
1267 \@chapapp\thechapter\@chappos\hskip1zw
1268 \fi
1269 (book) \fi
1270 \fi
1271 ##1}{}}%
1272 \def\sectionmark##1{\markright{%
1273 \ifnum \c@secnumdepth >\z@ \thesection \hskip1zw\fi
1274 ##1}}}%
1275 \end{book|report}

```

最後は学会誌の場合です。

```

1276 \*jspf
1277 \def\ps@headings{%
1278 \def\@oddfoot{\normalfont\hfil\thepage\hfil}
1279 \def\@evenfoot{\normalfont\hfil\thepage\hfil}
1280 \def\@oddhead{\normalfont\hfil \@title \hfil}
1281 \def\@evenhead{\normalfont\hfil プラズマ・核融合学会誌 \hfil}
1282 \end{jspf}

```

\ps@myheadings myheadings ページスタイルではユーザが \markboth や \markright で柱を設定するため、ここでの定義は非常に簡単です。

[2004-01-17] 渡辺徹さんのパッチを適用しました。

```

1283 \def\ps@myheadings{%
1284 \let\@oddfoot\@empty\let\@evenfoot\@empty

```

```

1285 \def\@evenhead{%
1286     \if@mparswitch \hss \fi%
1287     \hbox to \fullwidth{\thepage\hfil\leftmark}%
1288     \if@mparswitch\else \hss \fi}%
1289 \def\@oddhead{%
1290     \hbox to \fullwidth{\rightmark\hfil\thepage}\hss}%
1291 \let\@mkboth\@gobbletwo
1292 \langle book | report \rangle \let\chaptermark\@gobble
1293 \let\sectionmark\@gobble
1294 \langle !book & !report \rangle \let\subsectionmark\@gobble
1295 }

```

## 8 文書のマークアップ

### 8.1 表題

`\title` これらは L<sup>A</sup>T<sub>E</sub>X 本体で次のように定義されています。ここではコメントアウトした形で示します。

```

\date 1296 % \newcommand*{\title}[1]{\gdef\@title{#1}}
1297 % \newcommand*{\author}[1]{\gdef\@author{#1}}
1298 % \newcommand*{\date}[1]{\gdef\@date{#1}}
1299 % \date{\today}

```

`\etitle` 某学会誌スタイルで使う英語のタイトル、英語の著者名、キーワード、メールアドレスです。

```

\author 1300 \langle *jspf \rangle
\keywords 1301 \newcommand*{\etitle}[1]{\gdef\@etitle{#1}}
1302 \newcommand*{\eauthor}[1]{\gdef\@eauthor{#1}}
1303 \newcommand*{\keywords}[1]{\gdef\@keywords{#1}}
1304 \newcommand*{\email}[1]{\gdef\authors@mail{#1}}
1305 \newcommand*{\AuthorsEmail}[1]{\gdef\authors@mail{author's e-mail:\ #1}}
1306 \langle /jspf \rangle

```

`\plainifnotempty` 従来の標準クラスでは、文書全体のページスタイルを `empty` にしても表題のあるページだけ `plain` になってしまうことがありました。これは `\maketitle` の定義中に `\thispagestyle{plain}` が入っているためです。この問題を解決するために、「全体のページスタイルが `empty` でないならこのページのスタイルを `plain` にする」という次の命令を作ることにします。

```

1307 \def\plainifnotempty{%
1308     \ifx \@oddhead \@empty
1309         \ifx \@oddfoot \@empty
1310             \else
1311                 \thispagestyle{plainfoot}%
1312             \fi
1313         \else
1314             \thispagestyle{plainhead}%
1315         \fi}

```

`\maketitle` 表題を出力します。著者名を出力する部分は、欧文の標準クラスファイルでは `\large`、和文のものでは `\Large` になっていましたが、ここでは `\large` にしました。

[2016-11-16] 新設された `nomag` および `nomag*` オプションの場合をデフォルト (`usemag` 相当) に合わせるため、`\smallskip` を `\jsc@smallskip` に置き換えました。`\smallskip` のままでは `nomag(*)` の場合にスケールしなくなり、レイアウトが変わってしまいます。

```

1316 <*article | book | report | kiyou>
1317 \if@titlepage
1318   \newcommand{\maketitle}{%
1319     \begin{titlepage}%
1320       \let\footnotesize\small
1321       \let\footnoterule\relax
1322       \let\footnote\thanks
1323       \null\vfil
1324       \if@slide
1325         {\footnotesize \@date}%
1326         \begin{center}
1327           \mbox{} \|[1zw]
1328           \large
1329           {\maybeblue\hrule height0\jsc@mpt depth2\jsc@mpt\relax}\par
1330           \jsc@smallskip
1331           \@title
1332           \jsc@smallskip
1333           {\maybeblue\hrule height0\jsc@mpt depth2\jsc@mpt\relax}\par
1334           \vfill
1335           {\small \@author}%
1336         \end{center}
1337       \else
1338         \vskip 60\jsc@mpt
1339         \begin{center}%
1340           {\LARGE \@title \par}%
1341           \vskip 3em%
1342           {\large
1343             \lineskip .75em
1344             \begin{tabular}[t]{c}%
1345               \@author
1346             \end{tabular}\par}%
1347           \vskip 1.5em
1348           {\large \@date \par}%
1349         \end{center}%
1350       \fi
1351     \par
1352     \@thanks\vfil\null
1353   \end{titlepage}%
1354   \setcounter{footnote}{0}%
1355   \global\let\thanks\relax
1356   \global\let\maketitle\relax
1357   \global\let\@thanks\@empty

```

```

1358 \global\let\@author\@empty
1359 \global\let\@date\@empty
1360 \global\let\@title\@empty
1361 \global\let\title\relax
1362 \global\let\author\relax
1363 \global\let\date\relax
1364 \global\let\and\relax
1365 }%
1366 \else
1367 \newcommand{\maketitle}{\par
1368 \beginingroup
1369 \renewcommand\thefootnote{\@fnsymbol\c@footnote}%
1370 \def\@makefnmark{\rlap{\@textsuperscript{\normalfont\@thefnmark}}}%
1371 \long\def\@makefntext##1{\advance\leftskip 3zw
1372 \parindent 1zw\noindent
1373 \llap{\@textsuperscript{\normalfont\@thefnmark}\hskip0.3zw}##1}%
1374 \if@twocolumn
1375 \ifnum \col@number=\@ne
1376 \maketitle
1377 \else
1378 \twocolumn[\maketitle]%
1379 \fi
1380 \else
1381 \newpage
1382 \global\@topnum\z@ % Prevents figures from going at top of page.
1383 \maketitle
1384 \fi
1385 \plainifnotempty
1386 \@thanks
1387 \endgroup
1388 \setcounter{footnote}{0}%
1389 \global\let\thanks\relax
1390 \global\let\maketitle\relax
1391 \global\let\@thanks\@empty
1392 \global\let\@author\@empty
1393 \global\let\@date\@empty
1394 \global\let\@title\@empty
1395 \global\let\title\relax
1396 \global\let\author\relax
1397 \global\let\date\relax
1398 \global\let\and\relax
1399 }

```

`\@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。

```

1400 \def\@maketitle{%
1401 \newpage\null
1402 \vskip 2em
1403 \begin{center}%
1404 \let\footnote\thanks

```

```

1405     {\LARGE \@title \par}%
1406     \vskip 1.5em
1407     {\large
1408       \lineskip .5em
1409       \begin{tabular}[t]{c}%
1410         \@author
1411       \end{tabular}\par}%
1412     \vskip 1em
1413     {\large \@date}%
1414   \end{center}%
1415   \par\vskip 1.5em
1416   <article | report | kiyou> \ifvoid\@abstractbox\else\centerline{\box\@abstractbox}\vskip1.5em\fi
1417   }
1418   \fi
1419   </article | book | report | kiyou>
1420   <*jspf>
1421   \newcommand{\maketitle}{\par
1422     \begin{group
1423       \renewcommand\thefootnote{\@fnsymbol\c@footnote}%
1424       \def\@makefnmark{\rlap{\@textsuperscript{\normalfont\@thefnmark}}}%
1425       \long\def\@makefntext##1{\advance\leftskip 3zw
1426         \parindent 1zw\noindent
1427         \llap{\@textsuperscript{\normalfont\@thefnmark}\hspace{0.3zw}##1}%
1428         \twocolumn[\@maketitle]}%
1429       \plainifnotempty
1430       \@thanks
1431     \end{group
1432     \setcounter{footnote}{0}%
1433     \global\let\thanks\relax
1434     \global\let\maketitle\relax
1435     \global\let\@thanks\@empty
1436     \global\let\@author\@empty
1437     \global\let\@date\@empty
1438     % \global\let\@title\@empty % \@title は柱に使う
1439     \global\let\title\relax
1440     \global\let\author\relax
1441     \global\let\date\relax
1442     \global\let\and\relax
1443     \ifx\authors@mail\@undefined\else%
1444       \def\@makefntext{\advance\leftskip 3zw \parindent -3zw}%
1445       \footnotetext[0]{\itshape\authors@mail}%
1446     }\fi
1447     \global\let\authors@mail\@undefined}
1448   \def\@maketitle{%
1449     \newpage\null
1450     \vskip 6em % used to be 2em
1451     \begin{center}
1452       \let\footnote\thanks
1453       \ifx\@title\@undefined\else\LARGE\headfont\@title\par\fi

```

```

1454 \lineskip .5em
1455 \ifx\@author\@undefined\else
1456 \vskip 1em
1457 \begin{tabular}[t]{c}%
1458 \@author
1459 \end{tabular}\par
1460 \fi
1461 \ifx\@etitle\@undefined\else
1462 \vskip 1em
1463 {\large \@etitle \par}%
1464 \fi
1465 \ifx\@eauthor\@undefined\else
1466 \vskip 1em
1467 \begin{tabular}[t]{c}%
1468 \@eauthor
1469 \end{tabular}\par
1470 \fi
1471 \vskip 1em
1472 \@date
1473 \end{center}
1474 \vskip 1.5em
1475 \centerline{\box\@abstractbox}
1476 \ifx\@keywords\@undefined\else
1477 \vskip 1.5em
1478 \centerline{\parbox{157\jsc@mmm}{\textsf{Keywords:}}\ \small\@keywords}}
1479 \fi
1480 \vskip 1.5em}
1481 \</jspf>

```

## 8.2 章・節

■構成要素 \@startsection マクロは 6 個の必須引数と、オプションとして \* と 1 個のオプション引数と 1 個の必須引数をとります。

\@startsection{名}{レベル}{字下げ}{前アキ}{後アキ}{スタイル}  
\*[別見出し]{見出し}

それぞれの引数の意味は次の通りです。

**名** ユーザレベルコマンドの名前です (例: section)。

**レベル** 見出しの深さを示す数値です (chapter=1, section=2, ...)。この数値が secnumdepth 以下のとき見出し番号を出力します。

**字下げ** 見出しの字下げ量です。

**前アキ** この値の絶対値が見出し上側の空きです。負の場合は、見出し直後の段落をインデントしません。

**後アキ** 正の場合は、見出しの下側の空きです。負の場合は、絶対値が見出しの右の空きです。

(見出しと同じ行から本文を始めます)。

**スタイル** 見出しの文字スタイルの設定です。

\* この \* 印がないと、見出し番号を付け、見出し番号のカウンタに 1 を加算します。

**別見出し** 目次や柱に出力する見出しです。

**見出し** 見出しです。

見出しの命令は通常 `\@startsection` とその最初の 6 個の引数として定義されます。

次は `\@startsection` の定義です。情報処理学会論文誌スタイルファイル (`ipsjcommon.sty`) を参考にさせていただきましたが、完全に行送りが `\baselineskip` の整数倍にならなくてもいいから前の行と重ならないようにしました。

```
1482 \def\@startsection#1#2#3#4#5#6{%
1483   \if@noskipsec \leavevmode \fi
1484   \par
1485 % 見出し上の空きを \@tempskipa にセットする
1486   \@tempskipa #4\relax
1487 % \@afterindent は見出し直後の段落を字下げするかどうかを表すスイッチ
1488   \if@english \@afterindentfalse \else \@afterindenttrue \fi
1489 % 見出し上の空きが負なら見出し直後の段落を字下げしない
1490   \ifdim \@tempskipa <\z@
1491     \@tempskipa -\@tempskipa \@afterindentfalse
1492   \fi
1493   \if@nobreak
1494 %   \everypar{\everyparhook}% これは間違い
1495     \everypar{}%
1496   \else
1497     \addpenalty\@secpenalty
1498 % 次の行は削除
1499 %   \addvspace\@tempskipa
1500 % 次の \noindent まで追加
1501     \ifdim \@tempskipa >\z@
1502       \if@slide\else
1503         \null
1504         \vspace*{-\baselineskip}%
1505       \fi
1506       \vskip\@tempskipa
1507     \fi
1508   \fi
1509   \noindent
1510 % 追加終わり
1511   \@ifstar
1512     {\@ssect{#3}{#4}{#5}{#6}}%
1513     {\@dblarg{\@sect{#1}{#2}{#3}{#4}{#5}{#6}}}
```

`\@sect` と `\@xsect` は、前のアキがちょうどゼロの場合にもうまくいくように、多少変えてあります。`\everyparhook` も挿入しています。

```
1514 \def\@sect#1#2#3#4#5#6[#7]#8{%
1515   \ifnum #2>\c@secnumdepth
```



```

1516     \let\@svsec\@empty
1517 \else
1518     \refstepcounter{#1}%
1519     \protected@edef\@svsec{\@secntformat{#1}\relax}%
1520 \fi
1521 % 見出し後の空きを \@tempskipa にセット
1522 \@tempskipa #5\relax
1523 % 条件判断の順序を入れ替えました
1524 \ifdim \@tempskipa<\z@
1525     \def\@svsechd{%
1526         #6{\hskip #3\relax
1527             \@svsec #8}%
1528         \csname #1mark\endcsname{#7}%
1529         \addcontentsline{toc}{#1}{%
1530             \ifnum #2>\c@secnumdepth \else
1531                 \protect\numberline{\csname the#1\endcsname}%
1532             \fi
1533             #7}}% 目次にフルネームを載せるなら #8
1534 \else
1535     \begingroup
1536     \interlinepenalty \@M % 下から移動
1537     #6{%
1538         \@hangfrom{\hskip #3\relax\@svsec}%
1539 %     \interlinepenalty \@M % 上に移動
1540         #8\@@par}%
1541     \endgroup
1542     \csname #1mark\endcsname{#7}%
1543     \addcontentsline{toc}{#1}{%
1544         \ifnum #2>\c@secnumdepth \else
1545             \protect\numberline{\csname the#1\endcsname}%
1546         \fi
1547         #7}% 目次にフルネームを載せるならここは #8
1548 \fi
1549 \@xsect{#5}}

```

二つ挿入した `\everyparhook` のうち後者が `\paragraph` 類の後で 2 回実行され、それ以降は前者が実行されます。

[2016-07-28] `slide` オプションと `twocolumn` オプションを同時に指定した場合の罫線の位置を微調整しました。

```

1550 \def\@xsect#1{%
1551 % 見出しの後ろの空きを \@tempskipa にセット
1552 \@tempskipa #1\relax
1553 % 条件判断の順序を変えました
1554 \ifdim \@tempskipa<\z@
1555     \@nbreakfalse
1556     \global\@noskipsectrue
1557     \everypar{%
1558         \if@noskipsec

```

```

1559      \global\@noskipsecfalse
1560      {\setbox\z@\lastbox}%
1561      \clubpenalty\@M
1562      \begingroup \@svsechd \endgroup
1563      \unskip
1564      \@tempskipa #1\relax
1565      \hskip -\@tempskipa
1566      \else
1567      \clubpenalty \@clubpenalty
1568      \everypar{\everyparhook}%
1569      \fi\everyparhook}%
1570  \else
1571  \par \nobreak
1572  \vskip \@tempskipa
1573  \@afterheading
1574  \fi
1575  \if@slide
1576  {\vskip\if@twocolumn-5\jsc@empt\else-6\jsc@empt\fi
1577   \maybeblue\hrule height0\jsc@empt depth1\jsc@empt
1578   \vskip\if@twocolumn 4\jsc@empt\else 7\jsc@empt\fi\relax}%
1579  \fi
1580  \par % 2000-12-18
1581  \ignorespaces}
1582 \def\@ssect#1#2#3#4#5{%
1583   \@tempskipa #3\relax
1584   \ifdim \@tempskipa<\z@
1585     \def\@svsechd{#4{\hskip #1\relax #5}}%
1586   \else
1587     \begingroup
1588     #4{%
1589       \@hangfrom{\hskip #1}%
1590       \interlinepenalty \@M #5\@@par}%
1591     \endgroup
1592   \fi
1593   \@xsect{#3}}

```

## ■柱関係の命令

`\chaptermark` `\...mark` の形の命令を初期化します (第 7 節参照)。`\chaptermark` 以外は L<sup>A</sup>T<sub>E</sub>X 本体で定義済みです。

```

\subsectionmark 1594 \newcommand*\chaptermark[1]{}
\subsubsectionmark 1595 % \newcommand*\sectionmark[1]{}
1596 % \newcommand*\subsectionmark[1]{}
\paragraphmark 1597 % \newcommand*\subsubsectionmark[1]{}
\subparagraphmark 1598 % \newcommand*\paragraphmark[1]{}
1599 % \newcommand*\subparagraphmark[1]{}

```

## ■カウンタの定義

`\c@secnumdepth` `secnumdepth` は第何レベルの見出しまで番号を付けるかを定めるカウンタです。

```
1600 (!book&!report) \setcounter{secnumdepth}{3}
1601 (book | report) \setcounter{secnumdepth}{2}
```

`\c@chapter` 見出し番号のカウンタです。`\newcounter` の第 1 引数が新たに作るカウンタです。これは

`\c@section` 第 2 引数が増加するたびに 0 に戻されます。第 2 引数は定義済みのカウンタです。

```
\c@subsection 1602 \newcounter{part}
1603 (book | report) \newcounter{chapter}
\c@subsubsection 1604 (book | report) \newcounter{section}[chapter]
1605 (!book&!report) \newcounter{section}
\c@paragraph 1606 \newcounter{subsection}[section]
1607 \newcounter{subsubsection}[subsection]
1608 \newcounter{paragraph}[subsubsection]
1609 \newcounter{subparagraph}[paragraph]
```

`\thepart` カウンタの値を出力する命令 `\the` 何々 を定義します。

`\thechapter` カウンタを出力するコマンドには次のものがあります。

<code>\thesection</code>	<code>\arabic{COUNTER}</code>	1, 2, 3, ...
<code>\thesubsection</code>	<code>\roman{COUNTER}</code>	i, ii, iii, ...
<code>\thesubsubsection</code>	<code>\Roman{COUNTER}</code>	I, II, III, ...
<code>\theparagraph</code>	<code>\alph{COUNTER}</code>	a, b, c, ...
<code>\thesubparagraph</code>	<code>\Alph{COUNTER}</code>	A, B, C, ...
	<code>\kansuji{COUNTER}</code>	一, 二, 三, ...

以下ではスペース節約のため @ の付いた内部表現を多用しています。

```
1610 \renewcommand{\thepart}{\@Roman\c@part}
1611 (!book&!report)% \renewcommand{\thesection}{\@arabic\c@section}
1612 (!book&!report) \renewcommand{\thesection}{\presectionname\@arabic\c@section\postsectionname}
1613 (!book&!report) \renewcommand{\thesubsection}{\@arabic\c@section.\@arabic\c@subsection}
1614 (*book | report)
1615 \renewcommand{\thechapter}{\@arabic\c@chapter}
1616 \renewcommand{\thesection}{\thechapter.\@arabic\c@section}
1617 \renewcommand{\thesubsection}{\thesection.\@arabic\c@subsection}
1618 (/book | report)
1619 \renewcommand{\thesubsubsection}{%
1620   \thesubsection.\@arabic\c@subsubsection}
1621 \renewcommand{\theparagraph}{%
1622   \thesubsubsection.\@arabic\c@paragraph}
1623 \renewcommand{\thesubparagraph}{%
1624   \theparagraph.\@arabic\c@subparagraph}
```

`\@chapapp` `\@chapapp` の初期値は `\prechaptername` (第) です。

`\@chappos` `\@chappos` の初期値は `\postchaptername` (章) です。

`\appendix` は `\@chapapp` を `\appendixname` に, `\@chappos` を空に再定義します。

[2003-03-02] `\@secapp` は外しました。

```
1625 (book | report) \newcommand{\@chapapp}{\prechaptername}
```

```
1626 <book | report>\newcommand{\@chappos}{\postchaptername}
```

■前付, 本文, 後付 本のうち章番号があるのが「本文」、それ以外が「前付」「後付」です。

\frontmatter ページ番号をローマ数字にし、章番号を付けないようにします。

[2017-03-05] \frontmatter と \mainmatter の2つの命令は、改丁または改ページした後で \pagenumbering{...} でノンブルを1にリセットします。長い間 \frontmatter は openany のときに単なる改ページとしていましたが、これではノンブルをリセットする際に偶奇逆転が起こる場合があります。openany かどうかに関らず奇数ページまで繰るように修正することで、問題を解消しました。実は、L<sup>A</sup>T<sub>E</sub>X の標準クラスでは 1998 年に修正されていた問題です (コミュニティ版 pL<sup>A</sup>T<sub>E</sub>X の標準クラス 2017/03/05 も参照)。

```
1627 (*book)
1628 \newcommand\frontmatter{%
1629   \pltx@cleartooddpage
1630   \@mainmatterfalse
1631   \pagenumbering{roman}}
```

\mainmatter ページ番号を算用数字にし、章番号を付けるようにします。

```
1632 \newcommand\mainmatter{%
1633   \pltx@cleartooddpage
1634   \@mainmattertrue
1635   \pagenumbering{arabic}}
```

\backmatter 章番号を付けないようにします。ページ番号の付け方は変わりません。

```
1636 \newcommand\backmatter{%
1637   \if@openleft
1638     \cleardoublepage
1639   \else\if@openright
1640     \cleardoublepage
1641   \else
1642     \clearpage
1643   \fi\fi
1644   \@mainmatterfalse}
1645 </book>
```

## ■部

\part 新しい部を始めます。

\secdef を使って見出しを定義しています。このマクロは二つの引数をとります。

\secdef{星なし}{星あり}

星なし \* のない形の定義です。

星あり \* のある形の定義です。

\secdef は次のようにして使います。

```

\def\chapter { ... \secdef \CMDA \CMDB }
\def\CMDA    [#1]#2{...} % \chapter[...]{...} の定義
\def\CMDB    #1{...}    % \chapter*{...} の定義

```

まず book と report のクラス以外です。

```

1646 <*\book&!report>
1647 \newcommand\part{%
1648   \if@noskipsec \leavevmode \fi
1649   \par
1650   \addvspace{4ex}%
1651   \if@english \@afterindentfalse \else \@afterindenttrue \fi
1652   \secdef\@part\@spart}
1653 </!\book&!report>

```

book および report クラスの場合は、少し複雑です。

```

1654 <*\book | report>
1655 \newcommand\part{%
1656   \if@openleft
1657     \cleardoublepage
1658   \else\if@openright
1659     \cleardoublepage
1660   \else
1661     \clearpage
1662   \fi\fi
1663   \thispagestyle{empty}% 欧文用標準スタイルでは plain
1664   \if@twocolumn
1665     \onecolumn
1666     \@restonecoltrue
1667   \else
1668     \@restonecolfalse
1669   \fi
1670   \null\vfil
1671   \secdef\@part\@spart}
1672 </\book | report>

```

`\@part` 部の見出しを出力します。`\bfseries` を `\headfont` に変えました。

book および report クラス以外では `secnumdepth` が `-1` より大きいとき部番号を付けます。

```

1673 <*\book&!report>
1674 \def\@part[#1]#2{%
1675   \ifnum \c@secnumdepth >\m@ne
1676     \refstepcounter{part}%
1677     \addcontentsline{toc}{part}{%
1678       \prepartname\thepart\postpartname\hspace{1zw}#1}%
1679   \else
1680     \addcontentsline{toc}{part}{#1}%
1681   \fi
1682   \markboth{}{}%
1683   {\parindent\z@

```

```

1684 \raggedright
1685 \interlinepenalty \@M
1686 \normalfont
1687 \ifnum \c@secnumdepth >\m@ne
1688 \Large\headfont\prepartname\thepart\postpartname
1689 \par\nobreak
1690 \fi
1691 \huge \headfont #2%
1692 \markboth{}{}\par}%
1693 \nobreak
1694 \vskip 3ex
1695 \@afterheading}
1696 \!/book&!report)

```

book および report クラスでは secnumdepth が -2 より大きいとき部番号を付けます。

```

1697 (*book | report)
1698 \def\@part[#1]#2{%
1699 \ifnum \c@secnumdepth >-2\relax
1700 \refstepcounter{part}%
1701 \addcontentsline{toc}{part}{%
1702 \prepartname\thepart\postpartname\hspace{1zw}#1}%
1703 \else
1704 \addcontentsline{toc}{part}{#1}%
1705 \fi
1706 \markboth{}{}\%
1707 {\centering
1708 \interlinepenalty \@M
1709 \normalfont
1710 \ifnum \c@secnumdepth >-2\relax
1711 \huge\headfont \prepartname\thepart\postpartname
1712 \par\vskip20\jsc@empt
1713 \fi
1714 \Huge \headfont #2\par}%
1715 \@endpart}
1716 \!/book | report)

```

\@spart 番号を付けない部です。

```

1717 (*!book&!report)
1718 \def\@spart#1{%
1719 \parindent \z@ \raggedright
1720 \interlinepenalty \@M
1721 \normalfont
1722 \huge \headfont #1\par}%
1723 \nobreak
1724 \vskip 3ex
1725 \@afterheading}
1726 \!/book&!report)
1727 (*book | report)
1728 \def\@spart#1{%

```

```

1729 \centering
1730 \interlinepenalty \@M
1731 \normalfont
1732 \Huge \headfont #1\par}%
1733 \@endpart}
1734 </book | report>

```

`\@endpart` `\@part` と `\@spart` の最後で実行されるマクロです。両面印刷のときは白ページを追加します。二段組のときには、二段組に戻します。

[2016-12-13] `openany` のときには白ページが追加されるのは変なので、その場合は追加しないようにしました。このバグは L<sup>A</sup>T<sub>E</sub>X では `classes.dtx` v1.4b (2000/05/19) で修正されています。

```

1735 <(*book | report>
1736 \def\@endpart{\vfil\newpage
1737 \if@twoside
1738 \if@openleft %% added (2017/02/24)
1739 \null\thispagestyle{empty}\newpage
1740 \else\if@openright %% added (2016/12/13)
1741 \null\thispagestyle{empty}\newpage
1742 \fi\fi %% added (2016/12/13, 2017/02/24)
1743 \fi
1744 \if@restonecol
1745 \twocolumn
1746 \fi}
1747 </book | report>

```

## ■章

`\chapter` 章の最初のページスタイルは、全体が `empty` でなければ `plain` にします。また、`\@topnum` を 0 にして、章見出しの上に図や表が来ないようにします。

```

1748 <(*book | report>
1749 \newcommand{\chapter}{%
1750 \if@openleft\cleardoublepage\else
1751 \if@openright\cleardoublepage\else\clearpage\fi\fi
1752 \plainifnotempty % 元: \thispagestyle{plain}
1753 \global\@topnum\z@
1754 \if@english \@afterindentfalse \else \@afterindenttrue \fi
1755 \secdef
1756 {\@omit@numberfalse\@chapter}%
1757 {\@omit@numbertrue\@schapter}}

```

`\@chapter` 章見出しを出力します。`secnumdepth` が 0 以上かつ `\@mainmatter` が真のとき章番号を出力します。

```

1758 \def\@chapter[#1]#2{%
1759 \ifnum \c@secnumdepth >\m@ne
1760 <book> \if@mainmatter
1761 \refstepcounter{chapter}%

```

```

1762 \typeout{\@chapapp\thechapter\@chappos}%
1763 \addcontentsline{toc}{chapter}%
1764 {\protect\numberline
1765 % {\if@english\thechapter\else\@chapapp\thechapter\@chappos\fi}%
1766 {\@chapapp\thechapter\@chappos}%
1767 #1}%
1768 (book) \else\addcontentsline{toc}{chapter}{#1}\fi
1769 \else
1770 \addcontentsline{toc}{chapter}{#1}%
1771 \fi
1772 \chaptermark{#1}%
1773 \addtocontents{lof}{\protect\addvspace{10\jsc@mpt}}%
1774 \addtocontents{lot}{\protect\addvspace{10\jsc@mpt}}%
1775 \if@twocolumn
1776 \@topnewpage[\@makechapterhead{#2}]%
1777 \else
1778 \@makechapterhead{#2}%
1779 \@afterheading
1780 \fi}

```

`\@makechapterhead` 実際に章見出しを組み立てます。`\bfseries` を `\headfont` に変えました。

```

1781 \def\@makechapterhead#1{%
1782 \vspace*{2\Cvs}% 欧文は 50pt
1783 {\parindent \z@ \raggedright \normalfont
1784 \ifnum \c@secnumdepth >\m@ne
1785 (book) \if@mainmatter
1786 \huge\headfont \@chapapp\thechapter\@chappos
1787 \par\nobreak
1788 \vskip \Cvs % 欧文は 20pt
1789 (book) \fi
1790 \fi
1791 \interlinepenalty\@M
1792 \Huge\headfont #1\par\nobreak
1793 \vskip 3\Cvs}} % 欧文は 40pt

```

`\@schapter` `\chapter*{...}` コマンドの本体です。`\chaptermark` を補いました。

```

1794 \def\@schapter#1{%
1795 \chaptermark{#1}%
1796 \if@twocolumn
1797 \@topnewpage[\@makeschapterhead{#1}]%
1798 \else
1799 \@makeschapterhead{#1}\@afterheading
1800 \fi}

```

`\@makeschapterhead` 番号なしの章見出しです。

```

1801 \def\@makeschapterhead#1{%
1802 \vspace*{2\Cvs}% 欧文は 50pt
1803 {\parindent \z@ \raggedright
1804 \normalfont

```



```

1805 \interlinepenalty\@M
1806 \Huge \headfont #1\par\nobreak
1807 \vskip 3\Cvs}} % 欧文は 40pt
1808 </book|report>

```

## ■下位レベルの見出し

`\section` 欧文版では `\@startsection` の第 4 引数を負にして最初の段落の字下げを禁止していますが、和文版では正にして字下げするようにしています。

段組のときはなるべく左右の段が狂わないように工夫しています。

```

1809 \if@twocolumn
1810 \newcommand{\section}{%
1811 \jspf\ifx\maketitle\relax\else\maketitle\fi
1812 \@startsection{section}{1}{\z@}%
1813 \!kiyou {0.6\Cvs}{0.4\Cvs}%
1814 \!kiyou {\Cvs}{0.5\Cvs}%
1815 % {\normalfont\large\headfont\@secapp}}
1816 {\normalfont\large\headfont\raggedright}}
1817 \else
1818 \newcommand{\section}{%
1819 \if@slide\clearpage\fi
1820 \@startsection{section}{1}{\z@}%
1821 {\Cvs \@plus.5\Cdp \@minus.2\Cdp}% 前アキ
1822 {.5\Cvs \@plus.3\Cdp}% 後アキ
1823 % {\normalfont\Large\headfont\@secapp}}
1824 {\normalfont\Large\headfont\raggedright}}
1825 \fi

```

`\subsection` 同上です。

```

1826 \if@twocolumn
1827 \newcommand{\subsection}{\@startsection{subsection}{2}{\z@}%
1828 {\z@}{\if@slide .4\Cvs \else \z@ \fi}%
1829 {\normalfont\normalsize\headfont}}
1830 \else
1831 \newcommand{\subsection}{\@startsection{subsection}{2}{\z@}%
1832 {\Cvs \@plus.5\Cdp \@minus.2\Cdp}% 前アキ
1833 {.5\Cvs \@plus.3\Cdp}% 後アキ
1834 {\normalfont\large\headfont}}
1835 \fi

```

`\subsubsection` [2016-07-22] slide オプション指定時に `\subsubsection` の文字列と罫線が重なる問題に対処しました (forum:1982)。

```

1836 \if@twocolumn
1837 \newcommand{\subsubsection}{\@startsection{subsubsection}{3}{\z@}%
1838 {\z@}{\if@slide .4\Cvs \else \z@ \fi}%
1839 {\normalfont\normalsize\headfont}}
1840 \else
1841 \newcommand{\subsubsection}{\@startsection{subsubsection}{3}{\z@}%

```

```

1842    {\Cvs \@plus.5\Cdp \@minus.2\Cdp}%
1843    {\if@slide .5\Cvs \@plus.3\Cdp \else \z@ \fi}%
1844    {\normalfont\normalsize\headfont}}
1845 \fi

```

`\paragraph` 見出しの後ろで改行されません。

`\jsParagraphMark` [2016-11-16] 従来は `\paragraph` の最初に出るマークを「■」に固定していましたが、このマークを変更可能にするため `\jsParagraphMark` というマクロに切り出しました。これで、たとえば

```
\renewcommand{\jsParagraphMark}{★}
```

とすれば「★」に変更できますし、マークを空にすることも容易です。なお、某学会クラスでは従来どおりマークは付きません。

```

1846 (!jspf) \newcommand{\jsParagraphMark}{■}
1847 \if@twocolumn
1848   \newcommand{\paragraph}{\@startsection{paragraph}{4}{\z@}%
1849     {\z@}{\if@slide .4\Cvs \else -1zw\fi}% 改行せず 1zw のアキ
1850   (jspf)   {\normalfont\normalsize\headfont}}
1851 (!jspf)   {\normalfont\normalsize\headfont\jsParagraphMark}}
1852 \else
1853   \newcommand{\paragraph}{\@startsection{paragraph}{4}{\z@}%
1854     {0.5\Cvs \@plus.5\Cdp \@minus.2\Cdp}%
1855     {\if@slide .5\Cvs \@plus.3\Cdp \else -1zw\fi}% 改行せず 1zw のアキ
1856   (jspf)   {\normalfont\normalsize\headfont}}
1857 (!jspf)   {\normalfont\normalsize\headfont\jsParagraphMark}}
1858 \fi

```

`\subparagraph` 見出しの後ろで改行されません。

```

1859 \if@twocolumn
1860   \newcommand{\subparagraph}{\@startsection{subparagraph}{5}{\z@}%
1861     {\z@}{\if@slide .4\Cvs \@plus.3\Cdp \else -1zw\fi}%
1862     {\normalfont\normalsize\headfont}}
1863 \else
1864   \newcommand{\subparagraph}{\@startsection{subparagraph}{5}{\z@}%
1865     {\z@}{\if@slide .5\Cvs \@plus.3\Cdp \else -1zw\fi}%
1866     {\normalfont\normalsize\headfont}}
1867 \fi

```

### 8.3 リスト環境

第  $k$  レベルのリストの初期化をするのが `\@listk` です ( $k = i, ii, iii, iv$ )。 `\@listk` は `\leftmargin` を `\leftmargink` に設定します。

`\leftmargini` 二段組であるかないかに応じてそれぞれ 2em, 2.5em でしたが、ここでは全角幅の 2 倍にしました。

[2002-05-11] 3zw に変更しました。

[2005-03-19] 二段組は 2zw に戻しました。

```
1868 \if@slide
1869   \setlength\leftmargini{1zw}
1870 \else
1871   \if@twocolumn
1872     \setlength\leftmargini{2zw}
1873   \else
1874     \setlength\leftmargini{3zw}
1875   \fi
1876 \fi
```

`\leftmarginii` ii, iii, iv は `\labelsep` とそれぞれ ‘(m)’, ‘vii.’, ‘M.’ の幅との和より大きくすること  
`\leftmarginiii` になっています。ここでは全角幅の整数倍に丸めました。

```
\leftmarginiv 1877 \if@slide
\leftmarginv 1878   \setlength\leftmarginii {1zw}
1879   \setlength\leftmarginiii{1zw}
\leftmarginvi 1880   \setlength\leftmarginiv {1zw}
1881   \setlength\leftmarginv  {1zw}
1882   \setlength\leftmarginvi {1zw}
1883 \else
1884   \setlength\leftmarginii {2zw}
1885   \setlength\leftmarginiii{2zw}
1886   \setlength\leftmarginiv {2zw}
1887   \setlength\leftmarginv  {1zw}
1888   \setlength\leftmarginvi {1zw}
1889 \fi
```

`\labelsep` `\labelsep` はラベルと本文の間の距離です。`\labelwidth` はラベルの幅です。これは二分  
`\labelwidth` に変えました。

```
1890 \setlength \labelsep {0.5zw} % .5em
1891 \setlength \labelwidth{\leftmargini}
1892 \addtolength\labelwidth{-\labelsep}
```

`\partopsep` リスト環境の前に空行がある場合、`\parskip` と `\topsep` に `\partopsep` を加えた値だけ  
縦方向の空白ができます。0 に改変しました。

```
1893 \setlength\partopsep{\z@} % {2\p@ \@plus 1\p@ \@minus 1\p@}
```

`\@beginparpenalty` リストや段落環境の前後、リスト項目間に挿入されるペナルティです。

```
\@endparpenalty 1894 \@beginparpenalty -\@lowpenalty
\@itempenalty 1895 \@endparpenalty -\@lowpenalty
1896 \@itempenalty -\@lowpenalty
```

`\@listi` `\@listi` は `\leftmargin`, `\parsep`, `\topsep`, `\itemsep` などのトップレベルの定義を  
`\@listI` 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  
中では小さい値に設定されます）。このため、`\normalsize` がすべてのパラメータを戻せる  
ように、`\@listI` で `\@listi` のコピーを保存します。元の値はかなり複雑ですが、ここで  
は簡素化してしまいました。特に最初と最後に行送りの半分の空きが入るようにしてありま

す。アスキーの標準スタイルではトップレベルの `itemize`, `enumerate` 環境でだけ最初と最後に行送りの半分の空きが入るようになっていました。

[2004-09-27] `\topsep` のグルー  $\begin{smallmatrix} +0.2 \\ -0.1 \end{smallmatrix}$  `\baselineskip` を思い切って外しました。

```
1897 \def\@listi{\leftmargin\leftmarginii
1898   \parsep \z@
1899   \topsep 0.5\baselineskip
1900   \itemsep \z@ \relax}
1901 \let\@listI\@listi
```

念のためパラメータを初期化します（実際には不要のようです）。

```
1902 \@listi
```

`\@listii` 第 2~6 レベルのリスト環境のパラメータの設定です。

```
\@listiii 1903 \def\@listii{\leftmargin\leftmarginii
\@listiv 1904   \labelwidth\leftmarginii \advance\labelwidth-\labelsep
1905   \topsep \z@
\@listv 1906   \parsep \z@
\@listvi 1907   \itemsep\parsep}
1908 \def\@listiii{\leftmargin\leftmarginiii
1909   \labelwidth\leftmarginiii \advance\labelwidth-\labelsep
1910   \topsep \z@
1911   \parsep \z@
1912   \itemsep\parsep}
1913 \def\@listiv {\leftmargin\leftmarginiv
1914   \labelwidth\leftmarginiv
1915   \advance\labelwidth-\labelsep}
1916 \def\@listv {\leftmargin\leftmarginv
1917   \labelwidth\leftmarginv
1918   \advance\labelwidth-\labelsep}
1919 \def\@listvi {\leftmargin\leftmarginvi
1920   \labelwidth\leftmarginvi
1921   \advance\labelwidth-\labelsep}
```

■**enumerate 環境** `enumerate` 環境はカウンタ `enumi`, `enumii`, `enumiii`, `enumiv` を使います。`enumn` は第  $n$  レベルの番号です。

`\theenumi` 出力する番号の書式を設定します。これらは L<sup>A</sup>T<sub>E</sub>X 本体 (`ltlists.dtx` 参照) で定義済みですが、ここでは表し方を変えています。`\@arabic`, `\@alph`, `\@roman`, `\@Alph` はそれぞれ算用数字, 小文字アルファベット, 小文字ローマ数字, 大文字アルファベットで番号を出力する命令です。

```
1922 \renewcommand{\theenumi}{\@arabic\c@enumi}
1923 \renewcommand{\theenumii}{\@alph\c@enumii}
1924 \renewcommand{\theenumiii}{\@roman\c@enumiii}
1925 \renewcommand{\theenumiv}{\@Alph\c@enumiv}
```

`\labelenumi` `enumerate` 環境の番号を出力する命令です。第 2 レベル以外は最後に欧文のピリオドが付きますが、これは好みに応じて取り払ってください。第 2 レベルの番号のかっこは和文用に  
`\labelenumiii`  
`\labelenumiv`

換え、その両側に入る余分なグルーを `\inhibitglue` で取り除いています。

```
1926 \newcommand{\labelenumi}{\theenumi.}
1927 \newcommand{\labelenumii}{\inhibitglue (\theenumii) \inhibitglue}
1928 \newcommand{\labelenumiii}{\theenumiii.}
1929 \newcommand{\labelenumiv}{\theenumiv.}
```

`\p@enumii` `\p@enumn` は `\ref` コマンドで `enumerate` 環境の第  $n$  レベルの項目が参照されるときに書式です。これも第 2 レベルは和文用かっこにしました。

```
\p@enumiv 1930 \renewcommand{\p@enumii}{\theenumi}
1931 \renewcommand{\p@enumiii}{\theenumi\inhibitglue (\theenumii) }
1932 \renewcommand{\p@enumiv}{\p@enumiii\theenumiii}
```

### ■itemize 環境

`\labelitemi` `itemize` 環境の第  $n$  レベルのラベルを作るコマンドです。

```
\labelitemii 1933 \newcommand{\labelitemi}{\textbullet}
1934 \newcommand{\labelitemii}{\normalfont\bfseries \textendash}
\labelitemiii 1935 \newcommand{\labelitemiii}{\textasteriskcentered}
\labelitemiv 1936 \newcommand{\labelitemiv}{\textperiodcentered}
```

### ■description 環境

`description` 本来の `description` 環境では、項目名が短いと、説明部分の頭がそれに引きずられて左に出てしまいます。これを解決した新しい `description` の実装です。

```
1937 \newenvironment{description}{%
1938   \list{}{%
1939     \labelwidth=\leftmargin
1940     \labelsep=1zw
1941     \advance \labelwidth by -\labelsep
1942     \let \makelabel=\descriptionlabel}}{\endlist}
```

`\descriptionlabel` `description` 環境のラベルを出力するコマンドです。好みに応じて #1 の前に適当な空き（たとえば `\hspace{1zw}`）を入れるのもいいと思います。

```
1943 \newcommand*\descriptionlabel[1]{\normalfont\headfont #1\hfil}
```

### ■概要

`abstract` 概要（要旨、梗概）を出力する環境です。`book` クラスでは各章の初めにちょっとしたことを書くのに使います。`titlepage` オプション付きの `article` クラスでは、独立したページに出力されます。`abstract` 環境は元は `quotation` 環境で作られていましたが、`quotation` 環境の右マージンをゼロにしたので、`list` 環境で作り直しました。

JSPF スタイルでは実際の出力は `\maketitle` で行われます。

```
1944 (*book)
1945 \newenvironment{abstract}{%
1946   \begin{list}{}{%
1947     \listparindent=1zw
```

```

1948     \itemindent=\listparindent
1949     \rightmargin=0pt
1950     \leftmargin=5zw\item[]\{\end{list}\vspace{\baselineskip}}
1951 \end{book}
1952 \if*article|report|kiyou
1953 \newbox\@abstractbox
1954 \if@titlepage
1955     \newenvironment{abstract}{%
1956         \titlepage
1957         \null\vfil
1958         \@beginparpenalty\@lowpenalty
1959         \begin{center}%
1960             \headfont \abstractname
1961             \@endparpenalty\@M
1962         \end{center}}%
1963     {\par\vfil\null\endtitlepage}
1964 \else
1965     \newenvironment{abstract}{%
1966         \if@twocolumn
1967             \ifx\maketitle\relax
1968                 \section*{\abstractname}%
1969             \else
1970                 \global\setbox\@abstractbox\hbox\bgroup
1971                 \begin{minipage}[b]{\textwidth}
1972                     \small\parindent1zw
1973                     \begin{center}%
1974                         {\headfont \abstractname\vspace{-.5em}\vspace{\z@}}%
1975                     \end{center}%
1976                     \list{}{%
1977                         \listparindent\parindent
1978                         \itemindent \listparindent
1979                         \rightmargin \leftmargin}%
1980                     \item\relax
1981                 \fi
1982             \else
1983                 \small
1984                 \begin{center}%
1985                     {\headfont \abstractname\vspace{-.5em}\vspace{\z@}}%
1986                 \end{center}%
1987                 \list{}{%
1988                     \listparindent\parindent
1989                     \itemindent \listparindent
1990                     \rightmargin \leftmargin}%
1991                 \item\relax
1992             \fi\if@twocolumn
1993                 \ifx\maketitle\relax
1994                 \else
1995                     \endlist\end{minipage}\egroup
1996                 \fi

```

```

1997      \else
1998      \endlist
1999      \fi}
2000 \fi
2001 </article | report | kiyou>
2002 <*jspf>
2003 \newbox\@abstractbox
2004 \newenvironment{abstract}{%
2005     \global\setbox\@abstractbox\hbox\bgroup
2006     \begin{minipage}[b]{157\jsc@mmm}{\sffamily Abstract}\par
2007         \small
2008         \if@english \parindent6\jsc@mmm \else \parindent1zw \fi}%
2009     {\end{minipage}\egroup}
2010 </jspf>

```

### ■キーワード

**keywords** キーワードを準備する環境です。実際の出力は `\maketitle` で行われます。

```

2011 <*jspf>
2012 %\newbox\@keywordsbox
2013 %\newenvironment{keywords}{%
2014 %    \global\setbox\@keywordsbox\hbox\bgroup
2015 %    \begin{minipage}[b]{1570\jsc@mmm}{\sffamily Keywords:}\par
2016 %        \small\parindent0zw}%
2017 %    {\end{minipage}\egroup}
2018 </jspf>

```

### ■verse 環境

**verse** 詩のための `verse` 環境です。

```

2019 \newenvironment{verse}{%
2020     \let \=\@centercr
2021     \list{}{%
2022         \itemsep \z@
2023         \itemindent -2zw % 元: -1.5em
2024         \listparindent\itemindent
2025         \rightmargin \z@
2026         \advance\leftmargin 2zw}% 元: 1.5em
2027     \item\relax}{\endlist}

```

### ■quotation 環境

**quotation** 段落の頭の字下げ量を 1.5em から `\parindent` に変えました。また、右マージンを 0 にしました。

```

2028 \newenvironment{quotation}{%
2029     \list{}{%
2030         \listparindent\parindent
2031         \itemindent\listparindent

```

```

2032 \rightmargin \z@}%
2033 \item\relax}{\endlist}

```

## ■quote 環境

quote 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。

```

2034 \newenvironment{quote}%
2035 {\list{}{\rightmargin\z@}\item\relax}{\endlist}

```

■定理など ltthm.dtx 参照。たとえば次のように定義します。

```

\newtheorem{definition}{定義}
\newtheorem{axiom}{公理}
\newtheorem{theorem}{定理}

```

[2001-04-26] 定理の中はイタリック体になりましたが、これでは和文がゴシック体になってしまうので、\itshape を削除しました。

[2009-08-23] \bfseries を \headfont に直し、\labelsep を 1zw にし、括弧を全角にしました。

```

2036 \def\@begintheorem#1#2{\trivlist\labelsep=1zw
2037 \item[\hskip \labelsep{\headfont #1\ #2}]}
2038 \def\@opargbegintheorem#1#2#3{\trivlist\labelsep=1zw
2039 \item[\hskip \labelsep{\headfont #1\ #2 (#3) }]}

```

titlepage タイトルを独立のページに出力するのに使われます。

[2017-02-24] コミュニティ版 pL<sup>A</sup>T<sub>E</sub>X の標準クラス 2017/02/15 に合わせて、book クラスでタイトルを必ず奇数ページに送るようにしました。といっても、横組クラスしかありませんでしたので、従来の挙動は何も変わっていません。また、book 以外の場合のページ番号のリセットもコミュニティ版 pL<sup>A</sup>T<sub>E</sub>X の標準クラス 2017/02/15 に合わせましたが、こちらも片面印刷あるいは独立のタイトルページを作らないクラスばかりでしたので、従来の挙動は何も変わらずに済みました。

```

2040 \newenvironment{titlepage}{%
2041 (book) \pltx@cleartooddpage %% 2017-02-24
2042 \if@twocolumn
2043 \@restonecoltrue\onecolumn
2044 \else
2045 \@restonecolfalse\newpage
2046 \fi
2047 \thispagestyle{empty}%
2048 \ifodd\c@page\setcounter{page}\@ne\else\setcounter{page}\z@\fi %% 2017-02-24
2049 }%
2050 {\if@restonecol\twocolumn \else \newpage \fi
2051 \if@twoside\else
2052 \setcounter{page}\@ne
2053 \fi}

```

## ■付録



`\appendix` 本文と付録を分離するコマンドです。

```
2054 (*!book&!report)
2055 \newcommand{\appendix}{\par
2056   \setcounter{section}{0}%
2057   \setcounter{subsection}{0}%
2058   \gdef\presectionname{\appendixname}%
2059   \gdef\postsectionname{}}%
2060 % \gdef\thesection{\@Alph\c@section}% [2003-03-02]
2061 \gdef\thesection{\presectionname\@Alph\c@section\postsectionname}%
2062 \gdef\thesubsection{\@Alph\c@section.\@arabic\c@subsection}}
2063 \!/book&!report)
2064 (*book | report)
2065 \newcommand{\appendix}{\par
2066   \setcounter{chapter}{0}%
2067   \setcounter{section}{0}%
2068   \gdef\@chapapp{\appendixname}%
2069   \gdef\@chappos{}}%
2070 \gdef\thechapter{\@Alph\c@chapter}}
2071 \!/book | report)
```

## 8.4 パラメータの設定

### ■array と tabular 環境

`\arraycolsep` array 環境の列間には `\arraycolsep` の 2 倍の幅の空きが入ります。

```
2072 \setlength\arraycolsep{5\jsc@empt}
```

`\tabcolsep` tabular 環境の列間には `\tabcolsep` の 2 倍の幅の空きが入ります。

```
2073 \setlength\tabcolsep{6\jsc@empt}
```

`\arrayrulewidth` array, tabular 環境内の罫線の幅です。

```
2074 \setlength\arrayrulewidth{.4\jsc@empt}
```

`\doublerulesep` array, tabular 環境での二重罫線間のアキです。

```
2075 \setlength\doublerulesep{2\jsc@empt}
```

### ■tabbing 環境

`\tabbingsep` \’ コマンドで入るアキです。

```
2076 \setlength\tabbingsep{\labelsep}
```

### ■minipage 環境

`\@mpfootins` minipage 環境の脚注の `\skip\@mpfootins` は通常のページの `\skip\footins` と同じ働きをします。

```
2077 \skip\@mpfootins = \skip\footins
```

## ■framebox 環境

`\fboxsep` `\fbox`, `\framebox` で内側のテキストと枠との間の空きです。

`\fboxrule` `\fbox`, `\framebox` の罫線の幅です。

```
2078 \setlength\fboxsep{3\jsc@mp}{
2079 \setlength\fboxrule{.4\jsc@mp}
```

## ■equation と eqnarray 環境

`\theequation` 数式番号を出力するコマンドです。

```
2080 (!book&!report)\renewcommand \theequation {\@arabic\c@equation}
2081 (*book|report)
2082 \@addtoreset{equation}{chapter}
2083 \renewcommand\theequation
2084 {\ifnum \c@chapter>z@ \thechapter.\fi \@arabic\c@equation}
2085 </book|report)
```

`\jot` `eqnarray` の行間に余分に入るアキです。デフォルトの値をコメントアウトして示しておきます。

```
2086 % \setlength\jot{3pt}
```

`\@eqnnum` 数式番号の形式です。デフォルトの値をコメントアウトして示しておきます。

`\inhibitglue` (`\theequation`) `\inhibitglue` のように和文かっこを使うことも可能です。

```
2087 % \def\@eqnnum{(\theequation)}
```

`amsmath` パッケージを使う場合は `\tagform@` を次のように修正します。

```
2088 % \def\tagform@#1{\maketag@@@{(\ignorespaces#1\unskip\@italiccorr )}}
```

## 8.5 フロート

タイプ `TYPE` のフロートオブジェクトを扱うには、次のマクロを定義します。

`\fps@TYPE` フロートを置く位置 (float placement specifier) です。

`\ftype@TYPE` フロートの番号です。2 の累乗 (1, 2, 4, ...) でなければなりません。

`\ext@TYPE` フロートの目次を出力するファイルの拡張子です。

`\fnum@TYPE` キャプション用の番号を生成するマクロです。

`\@makecaption<num><text>` キャプションを出力するマクロです。`<num>` は `\fnum@...` の生成する番号、`<text>` はキャプションのテキストです。テキストは適当な幅の `\parbox` に入ります。

## ■figure 環境

`\c@figure` 図番号のカウンタです。

`\thefigure` 図番号を出力するコマンドです。

```
2089 <*\book&!report>
2090 \newcounter{figure}
2091 \renewcommand \thefigure {\@arabic\c@figure}
2092 <\/!\book&!report>
2093 <*\book | report>
2094 \newcounter{figure}[chapter]
2095 \renewcommand \thefigure
2096     {\ifnum \c@chapter>z@ \thechapter.\fi \@arabic\c@figure}
2097 <\/book | report>
```

`\fps@figure` `figure` のパラメータです。`\figurename` の直後に `~` が入っていましたが、ここでは外し  
`\ftype@figure` ました。

```
\ext@figure 2098 \def\fps@figure{tbp}
2099 \def\ftype@figure{1}
\fnun@figure 2100 \def\ext@figure{lof}
2101 \def\fnun@figure{\figurename\nobreak\thefigure}
```

`figure` \* 形式は段抜きのフロートです。

```
figure* 2102 \newenvironment{figure}%
2103     {\@float{figure}}%
2104     {\end@float}
2105 \newenvironment{figure*}%
2106     {\@dblfloat{figure}}%
2107     {\end@dblfloat}
```

## ■table 環境

`\c@table` 表番号カウンタと表番号を出力するコマンドです。アスキー版では `\thechapter.` が  
`\thetable` `\thechapter{}`・になっていますが、ここではオリジナルのままにしています。

```
2108 <*\book&!report>
2109 \newcounter{table}
2110 \renewcommand\thetable{\@arabic\c@table}
2111 <\/!\book&!report>
2112 <*\book | report>
2113 \newcounter{table}[chapter]
2114 \renewcommand \thetable
2115     {\ifnum \c@chapter>z@ \thechapter.\fi \@arabic\c@table}
2116 <\/book | report>
```

`\fps@table` `table` のパラメータです。`\tablename` の直後に `~` が入っていましたが、ここでは外しま  
`\ftype@table` した。

```
\ext@table 2117 \def\fps@table{tbp}
2118 \def\ftype@table{2}
\fnun@table 2119 \def\ext@table{lot}
2120 \def\fnun@table{\tablename\nobreak\thetable}
```

`table` \* は段抜きのフロートです。

`table*`

```

2121 \newenvironment{table}%
2122         {\@float{table}}%
2123         {\end@float}
2124 \newenvironment{table*}%
2125         {\@dblfloat{table}}%
2126         {\end@dblfloat}

```

## 8.6 キャプション

`\@makecaption` `\caption` コマンドにより呼び出され、実際にキャプションを出力するコマンドです。第 1 引数はフロートの番号、第 2 引数はテキストです。

`\abovecaptionskip` それぞれキャプションの前後に挿入されるスペースです。`\belowcaptionskip` が 0 になっていましたので、キャプションを表の上につけた場合にキャプションと表がくっついてしまうのを直しました。

```

2127 \newlength\abovecaptionskip
2128 \newlength\belowcaptionskip
2129 \setlength\abovecaptionskip{5\jsc@mp} % 元: 10\p@
2130 \setlength\belowcaptionskip{5\jsc@mp} % 元: 0\p@

```

実際のキャプションを出力します。オリジナルと異なり、文字サイズを `\small` にし、キャプションの幅を 2cm 狭くしました。

[2003-11-05] ロジックを少し変えてみました。

[2018-12-11] 遅くなりましたが、`listings` パッケージを使うときに `title` を指定すると“1zw”が出力されてしまう問題 (forum:1543, Issue #71) に対処しました。

```

2131 (*!jpf)
2132 % \long\def\@makecaption#1#2{\small
2133 %   \advance\leftskip10\jsc@mmm
2134 %   \advance\rightskip10\jsc@mmm
2135 %   \vskip\abovecaptionskip
2136 %   \sbox\@tempboxa{#1\hskip1zw\relax #2}%
2137 %   \ifdim \wd\@tempboxa >\hsize
2138 %     #1\hskip1zw\relax #2\par
2139 %   \else
2140 %     \global \minipagefalse
2141 %     \hb@xt@\hsize{\hfil\box\@tempboxa\hfil}%
2142 %   \fi
2143 %   \vskip\belowcaptionskip}}
2144 \long\def\@makecaption#1#2{\small
2145   \advance\leftskip .0628\linewidth
2146   \advance\rightskip .0628\linewidth
2147   \vskip\abovecaptionskip
2148   \sbox\@tempboxa{#1\hskip1zw\relax #2}%
2149   \ifdim \wd\@tempboxa <\hsize \centering \fi
2150   #1{\hskip1zw\relax}#2\par
2151   \vskip\belowcaptionskip}}

```

```

2152 </!jspf>
2153 <*jspf>
2154 \long\def\@makecaption#1#2{%
2155   \vskip\abovecaptionskip
2156   \sbox\@tempboxa{\small\sffamily #1\quad #2}%
2157   \ifdim \wd\@tempboxa >\hsize
2158     {\small\sffamily
2159      \list{#1}{%
2160        \renewcommand{\makelabel}[1]{##1\hfil}
2161        \itemsep \z@
2162        \itemindent \z@
2163        \labelsep \z@
2164        \labelwidth 11\jsc@mmm
2165        \listparindent\z@
2166        \leftmargin 11\jsc@mmm}\item\relax #2\endlist}
2167   \else
2168     \global \@minipagefalse
2169     \hb@xt@\hsize{\hfil\box\@tempboxa\hfil}%
2170   \fi
2171   \vskip\belowcaptionskip}
2172 </!jspf>

```

## 9 フォントコマンド

ここでは L<sup>A</sup>T<sub>E</sub>X 2.09 で使われていたコマンドを定義します。これら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のもので、できるだけ `\text...` と `\math...` を使ってください。

`\mc` フォントファミリーを変更します。

```

\gt 2173 \DeclareOldFontCommand{\mc}{\normalfont\mcfamily}{\mathmc}
\rm 2174 \DeclareOldFontCommand{\gt}{\normalfont\gtfamily}{\mathgt}
\sf 2175 \DeclareOldFontCommand{\rm}{\normalfont\rmfamily}{\mathrm}
\sf 2176 \DeclareOldFontCommand{\sf}{\normalfont\sffamily}{\mathsf}
\tt 2177 \DeclareOldFontCommand{\tt}{\normalfont\ttfamily}{\mathtt}

```

`\bf` ボールドシリーズにします。通常のミディアムシリーズに戻るコマンドは `\mdseries` です。

```

2178 \DeclareOldFontCommand{\bf}{\normalfont\bfseries}{\mathbf}

```

`\it` フォントシェイプを変えるコマンドです。斜体とスモールキャプスは数式中では何もしま  
`\sl` せん（警告メッセージを出力します）。通常のアップライト体に戻るコマンドは `\upshape`  
`\sc` です。

```

2179 \DeclareOldFontCommand{\it}{\normalfont\itshape}{\mathit}
2180 \DeclareOldFontCommand{\sl}{\normalfont\slshape}{\@nomath\sl}
2181 \DeclareOldFontCommand{\sc}{\normalfont\scshape}{\@nomath\sc}

```

`\cal` 数式モード以外では何もしません（警告を出します）。

`\mit`

```
2182 \DeclareRobustCommand*\cal{\@fontswitch\relax\mathcal}
2183 \DeclareRobustCommand*\mit{\@fontswitch\relax\mathnormal}
```

## 10 相互参照

### 10.1 目次の類

`\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}{タイトル}{ページ}
```

たとえば `\section` に見出し番号が付く場合、上の「タイトル」は

```
\numberline{番号}{見出し}
```

となります。この「番号」は `\thesection` コマンドで生成された見出し番号です。

`figure` 環境の `\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{番号}{キャプション}{ページ}}
```

この「番号」は `\thefigure` コマンドで生成された図番号です。

`table` 環境も同様です。

`\contentsline{...}` は `\l@...` というコマンドを実行するので、あらかじめ `\l@chapter`, `\l@section`, `\l@figure`などを定義しておかなければなりません。これらの多くは `\@dottedtocline` コマンドを使って定義します。これは

```
\@dottedtocline{レベル}{インデント}{幅}{タイトル}{ページ}
```

という書式です。

**レベル** この値が `tocdepth` 以下のときだけ出力されます。`\chapter` はレベル 0, `\section` はレベル 1, 等々です。

**インデント** 左側の字下げ量です。

**幅** 「タイトル」に `\numberline` コマンドが含まれる場合、節番号が入る箱の幅です。

`\@pnumwidth` ページ番号の入る箱の幅です。

`\@tocrmarg` 右マージンです。`\@tocrmarg ≥ \@pnumwidth` とします。

`\@dotsep` 点の間隔です (単位 `mu`)。

`\c@tocdepth` 目次ページに出力する見出しレベルです。元は `article` で 3, その他で 2 でしたが、ここでは一つずつ減らしています。

```
2184 \newcommand\@pnumwidth{1.55em}
2185 \newcommand\@tocrmarg{2.55em}
2186 \newcommand\@dotsep{4.5}
2187 (!book&!report)\setcounter{tocdepth}{2}
2188 (book|report)\setcounter{tocdepth}{1}
```

## ■目次

`\tableofcontents` 目次を生成します。

`\jsc@tocl@width` [2013-12-30] `\prechaptername` などから見積もった目次のラベルの長さです。(by ts)

```

2189 \newdimen\jsc@tocl@width
2190 \newcommand{\tableofcontents}{%
2191   (*book | report)
2192   \settowidth\jsc@tocl@width{\headfont\prechaptername\postchaptername}%
2193   \settowidth\@tempdima{\headfont\appendixname}%
2194   \ifdim\jsc@tocl@width<\@tempdima \setlength\jsc@tocl@width{\@tempdima}\fi
2195   \ifdim\jsc@tocl@width<2zw \divide\jsc@tocl@width by 2 \advance\jsc@tocl@width 1zw\fi
2196   \if@twocolumn
2197     \@restonecoltrue\onecolumn
2198   \else
2199     \@restonecolfalse
2200   \fi
2201   \chapter*{\contentsname}%
2202   \@mkboth{\contentsname}{}%
2203   </book | report>
2204   (*!book&!report)
2205   \settowidth\jsc@tocl@width{\headfont\presectionname\postsectionname}%
2206   \settowidth\@tempdima{\headfont\appendixname}%
2207   \ifdim\jsc@tocl@width<\@tempdima\relax\setlength\jsc@tocl@width{\@tempdima}\fi
2208   \ifdim\jsc@tocl@width<2zw \divide\jsc@tocl@width by 2 \advance\jsc@tocl@width 1zw\fi
2209   \section*{\contentsname}%
2210   \@mkboth{\contentsname}{\contentsname}%
2211   </!book&!report>
2212   \@starttoc{toc}%
2213   (book | report) \if@restonecol\twocolumn\fi
2214 }
```

`\l@part` 部の目次です。

```

2215 \newcommand*{\l@part}[2]{%
2216   \ifnum \c@tocdepth >-2\relax
2217   (!book&!report) \addpenalty\@secpenalty
2218   (book | report) \addpenalty{-\@highpenalty}%
2219   \addvspace{2.25em \@plus\jsc@mpt}%
2220   \begingroup
2221     \parindent \z@
2222     % \@pnumwidth should be \@tocrmarg
2223     % \rightskip \@pnumwidth
2224     \rightskip \@tocrmarg
2225     \parfillskip -\rightskip
2226     {\leavevmode
2227       \large \headfont
2228       \setlength\@lnumwidth{4zw}%
2229       #1\hfil \hb@xt@\@pnumwidth{\hss #2}}\par
2230     \nobreak
```

```

2231 <book | report>    \global\@nobreaktrue
2232 <book | report>    \everypar{\global\@nobreakfalse\everypar{}}%
2233    \endgroup
2234    \fi}

```

\l@chapter 章の目次です。 \@lnumwidth を 4.683zw に増やしました。

[2013-12-30] \@lnumwidth を \jsc@tocl@width から決めるようにしてみました。(by ts)

```

2235 <*book | report>
2236 \newcommand*{\l@chapter}[2]{%
2237   \ifnum \c@tocdepth >\m@ne
2238     \addpenalty{-\@highpenalty}%
2239     \addvspace{1.0em \@plus\jsc@empt}%
2240 %   \vskip 1.0em \@plus\p@    % book.cls では↑がこうなっている
2241   \begingroup
2242     \parindent\z@
2243 %   \rightskip\@pnumwidth
2244     \rightskip\@tocrmarg
2245     \parfillskip-\rightskip
2246     \leavevmode\headfont
2247 %   \if@english\setlength\@lnumwidth{5.5em}\else\setlength\@lnumwidth{4.683zw}\fi
2248     \setlength\@lnumwidth{\jsc@tocl@width}\advance\@lnumwidth 2.683zw
2249     \advance\leftskip\@lnumwidth \hskip-\leftskip
2250     #1\nobreak\hfil\nobreak\hbox to\@pnumwidth{\hss#2}\par
2251     \penalty\@highpenalty
2252   \endgroup
2253   \fi}
2254 </book | report>

```

\l@section 節の目次です。

```

2255 <!*book&!report>
2256 \newcommand*{\l@section}[2]{%
2257   \ifnum \c@tocdepth >\z@
2258     \addpenalty{\@secpenalty}%
2259     \addvspace{1.0em \@plus\jsc@empt}%
2260   \begingroup
2261     \parindent\z@
2262 %   \rightskip\@pnumwidth
2263     \rightskip\@tocrmarg
2264     \parfillskip-\rightskip
2265     \leavevmode\headfont
2266 %   \setlength\@lnumwidth{4zw}% 元 1.5em [2003-03-02]
2267     \setlength\@lnumwidth{\jsc@tocl@width}\advance\@lnumwidth 2zw
2268     \advance\leftskip\@lnumwidth \hskip-\leftskip
2269     #1\nobreak\hfil\nobreak\hbox to\@pnumwidth{\hss#2}\par
2270   \endgroup
2271   \fi}
2272 </!*book&!report>

```



インデントと幅はそれぞれ 1.5em, 2.3em でしたが, 1zw, 3.683zw に変えました。

2273 <book|report> % \newcommand\*{\l@section}{\@dottedtocline{1}{1zw}{3.683zw}}

[2013-12-30] 上のインデントは \jsc@tocl@width から決めるようにしました。(by ts)

\l@section さらに下位レベルの目次項目の体裁です。あまり使ったことがありませんので, 要修正かも  
\l@subsubsection しれません。

\l@paragraph [2013-12-30] ここも \jsc@tocl@width から決めるようにしてみました。(by ts)

\l@subparagraph 2274 <!\*book&!report>

2275 % \newcommand\*{\l@section}{\@dottedtocline{2}{1.5em}{2.3em}}

2276 % \newcommand\*{\l@subsubsection}{\@dottedtocline{3}{3.8em}{3.2em}}

2277 % \newcommand\*{\l@paragraph}{\@dottedtocline{4}{7.0em}{4.1em}}

2278 % \newcommand\*{\l@subparagraph}{\@dottedtocline{5}{10em}{5em}}

2279 %

2280 % \newcommand\*{\l@section}{\@dottedtocline{2}{1zw}{3zw}}

2281 % \newcommand\*{\l@subsubsection}{\@dottedtocline{3}{2zw}{3zw}}

2282 % \newcommand\*{\l@paragraph}{\@dottedtocline{4}{3zw}{3zw}}

2283 % \newcommand\*{\l@subparagraph}{\@dottedtocline{5}{4zw}{3zw}}

2284 %

2285 \newcommand\*{\l@section}{%

2286 \@tempdima\jsc@tocl@width \advance\@tempdima -1zw

2287 \@dottedtocline{2}{\@tempdima}{3zw}}

2288 \newcommand\*{\l@subsubsection}{%

2289 \@tempdima\jsc@tocl@width \advance\@tempdima 0zw

2290 \@dottedtocline{3}{\@tempdima}{4zw}}

2291 \newcommand\*{\l@paragraph}{%

2292 \@tempdima\jsc@tocl@width \advance\@tempdima 1zw

2293 \@dottedtocline{4}{\@tempdima}{5zw}}

2294 \newcommand\*{\l@subparagraph}{%

2295 \@tempdima\jsc@tocl@width \advance\@tempdima 2zw

2296 \@dottedtocline{5}{\@tempdima}{6zw}}

2297 </\*!book&!report>

2298 <\*book|report>

2299 % \newcommand\*{\l@section}{\@dottedtocline{2}{3.8em}{3.2em}}

2300 % \newcommand\*{\l@subsubsection}{\@dottedtocline{3}{7.0em}{4.1em}}

2301 % \newcommand\*{\l@paragraph}{\@dottedtocline{4}{10em}{5em}}

2302 % \newcommand\*{\l@subparagraph}{\@dottedtocline{5}{12em}{6em}}

2303 \newcommand\*{\l@section}{%

2304 \@tempdima\jsc@tocl@width \advance\@tempdima -1zw

2305 \@dottedtocline{1}{\@tempdima}{3.683zw}}

2306 \newcommand\*{\l@subsubsection}{%

2307 \@tempdima\jsc@tocl@width \advance\@tempdima 2.683zw

2308 \@dottedtocline{2}{\@tempdima}{3.5zw}}

2309 \newcommand\*{\l@subsubsection}{%

2310 \@tempdima\jsc@tocl@width \advance\@tempdima 6.183zw

2311 \@dottedtocline{3}{\@tempdima}{4.5zw}}

2312 \newcommand\*{\l@paragraph}{%

2313 \@tempdima\jsc@tocl@width \advance\@tempdima 10.683zw

2314 \@dottedtocline{4}{\@tempdima}{5.5zw}}

```

2315 \newcommand*{\l@subparagraph}{%
2316         \@tempdima\jsc@tocl@width \advance\@tempdima 16.183zw
2317         \@dottedtocline{5}{\@tempdima}{6.5zw}}
2318 \</book|report>

```

`\numberline` 欧文版 L<sup>A</sup>T<sub>E</sub>X では `\numberline{...}` は幅 `\@tempdima` の箱に左詰めで出力する命令ですが、アスキー版では `\@tempdima` の代わりに `\@lnumwidth` という変数で幅を決めるように再定義しています。後続文字が全角か半角かでスペースが変わらないように `\hspace` を入れておきました。

```

2319 \newdimen\@lnumwidth
2320 \def\numberline#1{\hb@xt@\@lnumwidth{#1\hfil}\hspace{0pt}}

```

`\@dottedtocline` L<sup>A</sup>T<sub>E</sub>X 本体 (ltsect.dtx 参照) での定義と同じですが、`\@tempdima` を `\@lnumwidth` に変えています。

[2018-06-23] デフォルトでは . . . . . のようにベースラインになります。これを変更可能にするため、`\jsTocLine` というマクロに切り出しました。例えば、仮想ボディの中央 . . . . . に変更したい場合は

```
\renewcommand{\jsTocLine}{\leaders \hbox {\hss \cdot\hss}\hfill}
```

とします。

```

2321 \def\jsTocLine{\leaders\hbox{%
2322     $\m@th \mkern \@dotsep mu\hbox{.}\mkern \@dotsep mu$}\hfill}
2323 \def\@dottedtocline#1#2#3#4#5{\ifnum #1>\c@tocdepth \else
2324     \vskip \z@ \@plus.2\jsc@mpt
2325     {\leftskip #2\relax \rightskip \@tocrmarg \parfillskip -\rightskip
2326       \parindent #2\relax\@afterindenttrue
2327       \interlinepenalty\@M
2328       \leavevmode
2329       \@lnumwidth #3\relax
2330       \advance\leftskip \@lnumwidth \null\nobreak\hskip -\leftskip
2331       {#4}\nobreak
2332       \jsTocLine \nobreak\hb@xt@\@pnumwidth{%
2333         \hfil\normalfont \normalcolor #5}\par}\fi}

```

## ■ 図目次と表目次

`\listoffigures` 図目次を出力します。

```

2334 \newcommand{\listoffigures}{%
2335 \< *book|report>
2336     \if@twocolumn\@restonecoltrue\onecolumn
2337     \else\@restonecolfalse\fi
2338     \chapter*{\listfigurename}%
2339     \@mkboth{\listfigurename}{}%
2340 \</book|report>
2341 \< !book&!report>
2342     \section*{\listfigurename}%
2343     \@mkboth{\listfigurename}{\listfigurename}%

```

```

2344 </!book&!report>
2345 \starttoc{lof}%
2346 <book | report> \if@restonecol\twocolumn\fi
2347 }

```

`\l@figure` 図目次の項目を出力します。

```

2348 \newcommand*{\l@figure}{\@dottedtocline{1}{1zw}{3.683zw}}

```

`\listoftables` 表目次を出力します。

```

2349 \newcommand{\listoftables}{%
2350 *book | report>
2351 \if@twocolumn\@restonecoltrue\onecolumn
2352 \else\@restonecolfalse\fi
2353 \chapter*{\listtablename}%
2354 \@mkboth{\listtablename}{}%
2355 </book | report>
2356 <!*book&!report>
2357 \section*{\listtablename}%
2358 \@mkboth{\listtablename}{\listtablename}%
2359 </!book&!report>
2360 \starttoc{lot}%
2361 <book | report> \if@restonecol\twocolumn\fi
2362 }

```

`\l@table` 表目次は図目次と同じです。

```

2363 \let\l@table\l@figure

```

## 10.2 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。元は 1.5em でした。

```

2364 \newdimen\bibindent
2365 \setlength\bibindent{2zw}

```

`thebibliography` 参考文献リストを出力します。

```

2366 \newenvironment{thebibliography}[1]{%
2367 \global\let\presectionname\relax
2368 \global\let\postsectionname\relax
2369 <article | jspf> \section*{\refname}\@mkboth{\refname}{\refname}%
2370 <*kiyou>
2371 \vspace{1.5\baselineskip}
2372 \subsubsection*{\refname}\@mkboth{\refname}{\refname}%
2373 \vspace{0.5\baselineskip}
2374 </kiyou>
2375 <book | report> \chapter*{\bibname}\@mkboth{\bibname}{}%
2376 <book | report> \addcontentsline{toc}{chapter}{\bibname}%
2377 \list{\@biblabel{\@arabic{c@enumiv}}}%
2378 {\settowidth\labelwidth{\@biblabel{#1}}%
2379 \leftmargin\labelwidth

```

```

2380      \advance\leftmargin\labelsep
2381      \@openbib@code
2382      \usecounter{enumiv}%
2383      \let\p@enumiv\@empty
2384      \renewcommand\theenumiv{\@arabic\c@enumiv}}%
2385 (kiyou)  \small
2386  \sloppy
2387  \clubpenalty4000
2388  \@clubpenalty\clubpenalty
2389  \widowpenalty4000%
2390  \sfcode'\.\@m}
2391  {\def\@noitemerr
2392   {\@latex@warning{Empty 'thebibliography' environment}}}%
2393  \endlist}

```

`\newblock` `\newblock` はデフォルトでは小さなスペースを生成します。

```

2394 \newcommand{\newblock}{\hskip .11em\@plus.33em\@minus.07em}

```

`\@openbib@code` `\@openbib@code` はデフォルトでは何もしません。この定義は `openbib` オプションによって変更されます。

```

2395 \let\@openbib@code\@empty

```

`\@biblabel` `\bibitem[...]` のラベルを作ります。ltbibl.dtx の定義の半角 `[]` を全角 `□` に変え、余分なスペースが入らないように `\inhibitglue` ではさみました。とりあえずコメントアウトしておきますので、必要に応じて生かしてください。

```

2396 % \def\@biblabel#1{\inhibitglue [#1] \inhibitglue}

```

`\cite` 文献の番号を出力する部分は ltbibl.dtx で定義されていますが、コンマとカッコを和文

`\@cite` フォントにするには次のようにします。とりあえずコメントアウトしておきましたので、必

`\@citex` 要に応じて生かしてください。かつこの前後に入るグルーを `\inhibitglue` で取っていますので、オリジナル同様、Knuth<sup>~</sup>`\cite{knu}` のように半角空白で囲んでください。

```

2397 % \def\@citex[#1]#2{\leavevmode
2398 %   \let\@citea\@empty
2399 %   \@cite{\@for\@citeb:=#2\do
2400 %     {\@citea\def\@citea{, \inhibitglue\penalty\@m\ }%
2401 %       \edef\@citeb{\expandafter\@firstofone\@citeb\@empty}}%
2402 %     \if@files\immediate\write\@auxout{\string\citation{\@citeb}}\fi
2403 %     \@ifundefined{b@\@citeb}{\mbox{\normalfont\bfseries ?}}%
2404 %     \G@refundefinedtrue
2405 %     \@latex@warning
2406 %       {Citation '\@citeb' on page \thepage \space undefined}}%
2407 %     {\@cite\ofmt{\csname b@\@citeb\endcsname}}}\@citex{#1}}
2408 % \def\@cite#1#2{\inhibitglue [{#1\if@tempwa , #2\fi}] \inhibitglue}

```

引用番号を上ツキの 1) のようなスタイルにするには次のようにします。`\cite` の先頭に `\unskip` を付けて先行のスペース ( ~ も ) を帳消しにしています。

```

2409 % \DeclareRobustCommand\cite{\unskip
2410 %   \@ifnextchar [{\@tempwattrue\@citex}{\@tempwafalse\@citex[]}]

```

```

2411 % \def\@cite#1#2{${\hbox{\scriptsize{#1}\if@tempswa
2412 % , \inhibitglue\ #2\fi}) }}$}

```

### 10.3 索引

**theindex** 2〜3 段組の索引を作成します。最後が偶数ページのときにマージンがずれる現象を直しました (Thanks: 藤村さん)。

```

2413 \newenvironment{theindex}{% 索引を 3 段組で出力する環境
2414     \if@twocolumn
2415         \onecolumn\@restonecolfalse
2416     \else
2417         \clearpage\@restonecoltrue
2418     \fi
2419     \columnseprule.4pt \columnsep 2zw
2420     \ifx\multicols\@undefined
2421 <book | report>         \twocolumn[\@makeschapterhead{\indexname}%
2422 <book | report>         \addcontentsline{toc}{chapter}{\indexname}]%
2423 <!book&!report>        \def\presectionname{}\def\postsectionname{}%
2424 <!book&!report>        \twocolumn[\section*{\indexname}]%
2425     \else
2426         \ifdim\textwidth<\fullwidth
2427             \setlength{\evensidemargin}{\oddsidemargin}
2428             \setlength{\textwidth}{\fullwidth}
2429             \setlength{\linewidth}{\fullwidth}
2430 <book | report>         \begin{multicols}{3}[\chapter*{\indexname}%
2431 <book | report>         \addcontentsline{toc}{chapter}{\indexname}]%
2432 <!book&!report>        \def\presectionname{}\def\postsectionname{}%
2433 <!book&!report>        \begin{multicols}{3}[\section*{\indexname}]%
2434     \else
2435 <book | report>         \begin{multicols}{2}[\chapter*{\indexname}%
2436 <book | report>         \addcontentsline{toc}{chapter}{\indexname}]%
2437 <!book&!report>        \def\presectionname{}\def\postsectionname{}%
2438 <!book&!report>        \begin{multicols}{2}[\section*{\indexname}]%
2439     \fi
2440     \fi
2441 <book | report>         \@mkboth{\indexname}{}%
2442 <!book&!report>        \@mkboth{\indexname}{\indexname}%
2443     \plainifnotempty % \thispagestyle{plain}
2444     \parindent\z@
2445     \parskip\z@ \@plus .3\jcs@mpt\relax
2446     \let\item\@idxitem
2447     \raggedright
2448     \footnotesize\narrowbaselines
2449 }{
2450     \ifx\multicols\@undefined
2451         \if@restonecol\onecolumn\fi
2452     \else

```

```

2453     \end{multicols}
2454     \fi
2455     \clearpage
2456 }

```

`\@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。

```

\subitem 2457 \newcommand{\@idxitem}{\par\hangindent 4zw} % 元 40pt
\subsubitem 2458 \newcommand{\subitem}{\@idxitem \hspace*{2zw}} % 元 20pt
2459 \newcommand{\subsubitem}{\@idxitem \hspace*{3zw}} % 元 30pt

```

`\indexspace` 索引で先頭文字ごとのブロックの間に入るスペースです。

```

2460 \newcommand{\indexspace}{\par \vskip 10\jsc@mpt \@plus5\jsc@mpt \@minus3\jsc@mpt\relax}

```

`\seename` 索引の `\see`, `\seealso` コマンドで出力されるものです。デフォルトはそれぞれ *see*, *see also* という英語ですが, ここではとりあえず両方とも「→」に変えました。⇒ (`\Rightarrow`) などでもいいでしょう。

```

2461 \newcommand\seename{\if@english see\else →\fi}
2462 \newcommand\alsoname{\if@english see also\else →\fi}

```

## 10.4 脚注

`\footnote` 和文の句読点・閉じかっこ類の直後で用いた際に余分なアキが入るのを防ぐため, `\footnotemark` `\inhibitglue` を入れることにします。pL<sup>A</sup>T<sub>E</sub>X の日付が 2016/09/03 より新しい場合は, このパッチが不要なのであてません。

```

2463 \@ifl@t@r\pfmtversion{2016/09/03}
2464     {\jsc@needspace@tchfalse}{\jsc@needspace@tchtrue}
2465 \ifjsc@needspace@tch
2466     \let\footnotes@ve=\footnote
2467     \def\footnote{\inhibitglue\footnotes@ve}
2468     \let\footnotemarks@ve=\footnotemark
2469     \def\footnotemark{\inhibitglue\footnotemarks@ve}
2470 \fi

```

`\@makefnmark` 脚注番号を付ける命令です。ここでは脚注番号の前に記号 \* を付けています。「注 1」の形式にするには `\textasteriskcentered` を `注 \kern0.1em` にしてください。`\@xfootnotenext` と合わせて, もし脚注番号が空なら記号も出力しないようにしてあります。

[2002-04-09] インプリメントの仕方を変えたため消しました。

[2013-04-23] 新しい pT<sub>E</sub>X では脚注番号のまわりにスペースが入りすぎることを防ぐため, 北川さんのパッチ [qa:57090] を取り込みました。

[2013-05-14] plcore.ltx に倣った形に書き直しました (Thanks: 北川さん)。

[2016-07-11] コミュニティ版 pL<sup>A</sup>T<sub>E</sub>X の変更に追随しました (Thanks: 角藤さん)。pL<sup>A</sup>T<sub>E</sub>X の日付が 2016/04/17 より新しい場合は, このパッチが不要なのであてません。

```

2471 \@ifl@t@r\pfmtversion{2016/04/17}
2472     {\jsc@needspace@tchfalse}{\jsc@needspace@tchtrue}

```

```

2473 \ifjsc@needspace@tch
2474 \renewcommand\@makefnmark{%
2475   \ifydir \hbox{\hbox{\@textsuperscript{\normalfont\@thefnmark}}\hbox{}}%
2476   \else\hbox{\yoko\@textsuperscript{\normalfont\@thefnmark}}\fi}
2477 \fi

```

`\thefootnote` 脚注番号に \* 印が付くようにしました。ただし、番号がゼロのときは \* 印も脚注番号も付きません。

[2003-08-15] `\textasteriskcentered` ではフォントによって下がりすぎるので変更しました。

[2016-10-08] TODO: 脚注番号が `newttext` や `newpertext` の使用時におかしくなっています。これらのパッケージは内部で `\thefootnote` を再定義していますので、気になる場合はパッケージを読み込むときに `defaultsup` オプションを付けてください (qa:57284, qa:57287)。

```

2478 \def\thefootnote{\ifnum\c@footnote>\z@\leavevmode\lower.5ex\hbox{*}\@arabic\c@footnote\fi}

```

「注 1」の形式にするには次のようにしてください。

```

2479 % \def\thefootnote{\ifnum\c@footnote>\z@ 注 \kern0.1zw\@arabic\c@footnote\fi}

```

`\footnoterule` 本文と脚注の間の罫線です。

```

2480 \renewcommand{\footnoterule}{%
2481   \kern-3\jsc@empt
2482   \hrule width .4\columnwidth height 0.4\jsc@empt
2483   \kern 2.6\jsc@empt}

```

`\c@footnote` 脚注番号は章ごとにリセットされます。

```

2484 (book | report) \addtoreset{footnote}{chapter}

```

`\@footnotetext` 脚注で `\verb` が使えるように改変してあります。Jeremy Gibbons, *T<sub>E</sub>X and TUG NEWS*, Vol. 2, No. 4 (1993), p. 9)

[2016-08-25] コミュニティ版 p<sub>Λ</sub>T<sub>E</sub>X の「閉じ括弧類の直後に `\footnotetext` が続く場合に改行が起きることがある問題に対処」と同等のコードを追加しました。

[2016-09-08] コミュニティ版 p<sub>Λ</sub>T<sub>E</sub>X のバグ修正に追随しました。

[2016-11-29] 古い p<sub>Λ</sub>T<sub>E</sub>X で使用された場合を考慮してコードを改良。

[2018-03-11] `\next` などいくつかの内部命令を `\jsc@...` 付きのユニークな名前にしました。

```

2485 \long\def\@footnotetext{%
2486   \insert\footins\bgroup
2487     \normalfont\footnotesize
2488     \interlinepenalty\interfootnotelinepenalty
2489     \splittopskip\footnotesep
2490     \splitmaxdepth \dp\strutbox \floatingpenalty \MM
2491     \hsize\columnwidth \@parboxrestore
2492     \protected@edef\@currentlabel{%
2493       \csname p@footnote\endcsname\@thefnmark
2494     }%

```

```

2495 \color@begingroup
2496 \makefnfntext{%
2497 \rule{z@}{\footnotesep\ignorespaces}%
2498 \futurelet\jsc@next\jsc@fo@t}
2499 \def\jsc@fo@t{\ifcat\bgroup\noexpand\jsc@next \let\jsc@next\jsc@f@t
2500 \else \let\jsc@next\jsc@f@t\fi \jsc@next}
2501 \def\jsc@f@t{\bgroup\aftergroup\jsc@@foot\let\jsc@next}
2502 \def\jsc@f@t#1{#1\jsc@@foot}
2503 \def\jsc@@foot{\@finalstrut\strutbox\color@endgroup\egroup
2504 \ifx\pltx@foot@penalty\undefined\else
2505 \ifhmode\null\fi
2506 \ifnum\pltx@foot@penalty=z@\else
2507 \penalty\pltx@foot@penalty
2508 \pltx@foot@penalty\z@
2509 \fi
2510 \fi}

```

`\makefnfntext` 実際に脚注を出力する命令です。`\makefnmark` は脚注の番号を出力する命令です。ここでは脚注が左端から一定距離に来るようにしてあります。

```

2511 \newcommand\makefnfntext[1]{%
2512 \advance\leftskip 3zw
2513 \parindent 1zw
2514 \noindent
2515 \llap{\makefnmark\hskip0.3zw}#1}

```

`\xfootnotenext` 最初の `\footnotetext{...}` は番号が付きません。著者の所属などを脚注の欄に書くときに便利です。

すでに `\footnote` を使った後なら `\footnotetext[0]{...}` とすれば番号を付けない脚注になります。ただし、この場合は脚注番号がリセットされてしまうので、工夫が必要です。

[2002-04-09] インプリメントの仕方を変えたため消しました。

```

2516 % \def\xfootnotenext[#1]{%
2517 % \begingroup
2518 % \ifnum#1>z@
2519 % \csname c@\mpfn\endcsname #1\relax
2520 % \unrestored@protected\xdef\@thefnmark{\thempfn}%
2521 % \else
2522 % \unrestored@protected\xdef\@thefnmark{}%
2523 % \fi
2524 % \endgroup
2525 % \@footnotetext}

```

## 11 段落の頭へのグルー挿入禁止

段落頭のかぎカッコなどを見かけ 1 字半下げから全角 1 字下げに直します。

[2016-07-18] `\inhibitglue` の発行対象を `\inhibitxspcode` が 2 に設定されているも



のすべてに拡大しました。

[2016-12-01] すぐ上の変更で `\@tempa` を使っていたのがよくなかったので、プレフィックスを付けて `\jsc@tempa` にしました (forum:2085)。

[2017-02-13] `\jsc@tempa` は実はテンポラリではなく「この処理専用のユニーク制御綴」である必要があります。間違って別の箇所で使う危険性が高いので、専用の命令 `\jsc@ig@temp` に置き換えました (Issue #54)。

```
2526 \def\@inhibitglue{%
2527   \futurelet\@let@token\@inhibitglue}
2528 \begingroup
2529 \let\GDEF=\gdef
2530 \let\CATCODE=\catcode
2531 \let\ENDGROUP=\endgroup
2532 \CATCODE'k=12
2533 \CATCODE'a=12
2534 \CATCODE'n=12
2535 \CATCODE'j=12
2536 \CATCODE'i=12
2537 \CATCODE'c=12
2538 \CATCODE'h=12
2539 \CATCODE'r=12
2540 \CATCODE't=12
2541 \CATCODE'e=12
2542 \GDEF\KANJI@CHARACTER{kanji character }
2543 \ENDGROUP
2544 \def\@inhibitglue{%
2545   \expandafter\expandafter\expandafter\jsc@inhibitglue\expandafter\meaning\expandafter\@let@token
2546   \expandafter\def\expandafter\jsc@inhibitglue\expandafter#\expandafter1\KANJI@CHARACTER#2#3\jsc
2547   \def\jsc@ig@temp{#1}%
2548   \ifx\jsc@ig@temp\@empty
2549     \ifnum\the\inhibitxspcode'#2=2\relax
2550       \inhibitglue
2551     \fi
2552   \fi}
2553 \let\everyparhook=\@inhibitglue
2554 \AtBeginDocument{\everypar{\everyparhook}}
```

これだけではいけないようです。あちこちに `\everypar` を初期化するコマンドが隠されていました。

まず、環境の直後の段落です。

[2016-11-19] `ltlists.dtx` 2015/05/10 v1.0t の変更に従って `\clubpenalty` のリセットを追加しました。

```
2555 \def\@doendpe{%
2556   \@endpetrue
2557   \def\par{%
2558     \@restorepar\clubpenalty\@clubpenalty\everypar{\everyparhook}\par\@endpefalse}%
2559   \everypar{\setbox\z@\lastbox}\everypar{\everyparhook}\@endpefalse\everyparhook}}
```

[2017-08-31] minipage 環境にも対策します。

```
2560 \def\@setminipage{%  
2561   \@minipagetrue  
2562   \everypar{\@minipagefalse\everypar{\everyparhook}}%  
2563 }
```

\item 命令の直後です。

```
2564 \def\@item[#1]{%  
2565   \if@noperitem  
2566     \@donoperitem  
2567   \else  
2568     \if@inlabel  
2569       \indent \par  
2570     \fi  
2571     \ifhmode  
2572       \unskip\unskip \par  
2573     \fi  
2574     \if@newlist  
2575       \if@nobreak  
2576         \@nbitem  
2577       \else  
2578         \addpenalty\@beginparpenalty  
2579         \addvspace\@topsep  
2580         \addvspace{-\parskip}%  
2581       \fi  
2582     \else  
2583       \addpenalty\@itempenalty  
2584       \addvspace\itemsep  
2585     \fi  
2586     \global\@inlabeltrue  
2587   \fi  
2588   \everypar{%  
2589     \@minipagefalse  
2590     \global\@newlistfalse  
2591     \if@inlabel  
2592       \global\@inlabelfalse  
2593       {\setbox\z@\lastbox  
2594        \ifvoid\z@  
2595          \kern-\itemindent  
2596        \fi}%  
2597     \box\@labels  
2598     \penalty\z@  
2599   \fi  
2600   \if@nobreak  
2601     \@nobreakfalse  
2602     \clubpenalty \@M  
2603   \else  
2604     \clubpenalty \@clubpenalty  
2605     \everypar{\everyparhook}%
```

```

2606 \fi\everyparhook}%
2607 \if@noitemarg
2608 \@noitemargfalse
2609 \if@nmbrlist
2610 \refstepcounter\@listctr
2611 \fi
2612 \fi
2613 \sbox\@tempboxa{\makelabel{#1}}%
2614 \global\setbox\@labels\hbox{%
2615 \unhbox\@labels
2616 \hskip \itemindent
2617 \hskip -\labelwidth
2618 \hskip -\labelsep
2619 \ifdim \wd\@tempboxa >\labelwidth
2620 \box\@tempboxa
2621 \else
2622 \hbox to\labelwidth {\unhbox\@tempboxa}%
2623 \fi
2624 \hskip \labelsep}%
2625 \ignorespaces}

```

二つ挿入した `\everyparhook` のうち後者が `\section` 類の直後に 2 回、前者が 3 回目以降に実行されます。

```

2626 \def\@afterheading{%
2627 \@nobraektrue
2628 \everypar{%
2629 \if@nobraek
2630 \@nobraekfalse
2631 \clubpenalty \@M
2632 \if@afterindent \else
2633 {\setbox\z@\lastbox}%
2634 \fi
2635 \else
2636 \clubpenalty \@clubpenalty
2637 \everypar{\everyparhook}%
2638 \fi\everyparhook}}

```

`\@gnewline` についてはちょっと複雑な心境です。もともとの  $\text{p}\text{L}\text{A}\text{T}\text{E}\text{X} 2_{\epsilon}$  は段落の頭にグルーが入る方で統一されていました。しかし `\` の直後にはグルーが入らず、不統一でした。そこで `\` の直後にもグルーを入れるように直していただいた経緯があります。しかし、ここでは逆にグルーを入れない方で統一したいので、また元に戻してしまいました。

しかし単に戻すだけでも駄目みたいなので、ここでも最後にグルーを消しておきます。

```

2639 \def\@gnewline #1{%
2640 \ifvmode
2641 \@nolnerr
2642 \else
2643 \unskip \reserved@e {\reserved@f#1}\nobraek \hfil \break \null
2644 \inhibitglue \ignorespaces

```

```
2645 \fi}
```

## 12 いろいろなロゴ

L<sup>A</sup>T<sub>E</sub>X 関連のロゴを作り直します。

[2016-07-14] ロゴの定義は jslogo パッケージに移転しました。後方互換のため、jsclasses ではデフォルトでこれを読み込みます。nojslogo オプションが指定されている場合は読み込みません。

\small 文字を小さめに出したり上寄りに小さめに出したりする命令を、jslogo.sty では名称変更  
\small してありますので、コピーします。

```
2646 \if@jslogo
2647 \IfFileExists{jslogo.sty}{%
2648   \RequirePackage{jslogo}%
2649   \def\small{\jslg@small}%
2650   \def\small{\jslg@uppersmall}%
2651 }{%
2652   \ClassWarningNoLine{\jsc@clsname}{%
2653     The redefinitions of LaTeX-related logos has\MessageBreak
2654     been moved to jslogo.sty since 2016, but\MessageBreak
2655     jslogo.sty not found. Current release of\MessageBreak
2656     'jsclasses' includes it, so please check\MessageBreak
2657     the installation}%
2658 }
2659 \fi
```

## 13 amsmath との衝突の回避

\ltx@ifnextchar amsmath パッケージでは行列中で \@ifnextchar を再定義していますが、これが L<sup>A</sup>T<sub>E</sub>X の  
\ProvidesFile \ProvidesFile で悪さをする例が F<sup>T</sup>E<sub>X</sub> で報告されています。これを避けるための tDB  
さんのフィックスを挿入しておきます。副作用がありましたらお知らせください。

この現象については私の TeX 掲示板 4273～, 16058～ で議論がありました。なお、AMS  
関係のパッケージを読み込む際に psamsfonts オプションを与えても回避できます (Thanks:  
しっぽ愛好家さん)。

[2016-11-19] 本家の ltclass.dtx 2004/01/28 v1.1g で修正されているのでコメントアウト  
しました。

```
2660 %\let\ltx@ifnextchar\@ifnextchar
2661 %\def\ProvidesFile#1{%
2662 % \begingroup
2663 %   \catcode'\ 10 %
2664 %   \ifnum \endlinechar<256 %
2665 %     \ifnum \endlinechar>\m@ne
2666 %       \catcode\endlinechar 10 %
2667 %     \fi
```

```

2668 % \fi
2669 % \@makeother\/%
2670 % \@makeother\&%
2671 % \ltx@ifnextchar[{\@providesfile{#1}}{\@providesfile{#1}[]}]

```

## 14 初期設定

### ■いろいろな語

```

\prepartname
\postpartname 2672 \newcommand{\prepartname}{\if@english Part~\else 第\fi}
\prechaptername 2673 \newcommand{\postpartname}{\if@english\else 部\fi}
2674 (book|report)\newcommand{\prechaptername}{\if@english Chapter~\else 第\fi}
\postchaptername 2675 (book|report)\newcommand{\postchaptername}{\if@english\else 章\fi}
\presectionname 2676 \newcommand{\presectionname}{}% 第
\postsectionname 2677 \newcommand{\postsectionname}{}% 節

\contentsname
\listfigurename 2678 \newcommand{\contentsname}{\if@english Contents\else 目次\fi}
\listtablename 2679 \newcommand{\listfigurename}{\if@english List of Figures\else 図目次\fi}
2680 \newcommand{\listtablename}{\if@english List of Tables\else 表目次\fi}

\refname
\bibname 2681 \newcommand{\refname}{\if@english References\else 参考文献\fi}
\indexname 2682 \newcommand{\bibname}{\if@english Bibliography\else 参考文献\fi}
2683 \newcommand{\indexname}{\if@english Index\else 索引\fi}

\figurename
\tablename 2684 (!jspf)\newcommand{\figurename}{\if@english Fig.~\else 図\fi}
2685 (jspf)\newcommand{\figurename}{Fig.~}
2686 (!jspf)\newcommand{\tablename}{\if@english Table~\else 表\fi}
2687 (jspf)\newcommand{\tablename}{Table~}

\appendixname
\abstractname 2688 % \newcommand{\appendixname}{\if@english Appendix~\else 付録\fi}
2689 \newcommand{\appendixname}{\if@english\else 付録\fi}
2690 (!book)\newcommand{\abstractname}{\if@english Abstract\else 概要\fi}

\today
2691 \newif\if 西暦 \西暦 true
2692 \def\西暦{\西暦 true}
2693 \def\和暦{\西暦 false}
2694 \newcount\heisei \heisei\year \advance\heisei-1988\relax

```

■今日の日付  $\LaTeX$  で処理した日付を出力します。和暦にするには `\和暦` と書いてください。ちなみにこの文章の作成日は西暦では 2020 年 10 月 5 日で、和暦では令和 2 年 10 月 5 日です。

```

2695 \def\pltx@today@year@#1{%
2696   \ifnum\numexpr\year-#1=1 元 \else
2697     \ifnum1=\iftdir\ifmdir0\else1\fi\else0\fi
2698       \kansuji\numexpr\year-#1\relax
2699     \else
2700       \number\numexpr\year-#1\relax\nobreak
2701     \fi
2702   \fi 年
2703 }
2704 \def\pltx@today@year{%
2705   \ifnum\numexpr\year*10000+\month*100+\day<19890108
2706     昭和\pltx@today@year@{1925}%
2707   \else\ifnum\numexpr\year*10000+\month*100+\day<20190501
2708     平成\pltx@today@year@{1988}%
2709   \else
2710     令和\pltx@today@year@{2018}%
2711   \fi\fi}
2712 \def\today{%
2713   \if@english
2714     \ifcase\month\or
2715       January\or February\or March\or April\or May\or June\or
2716       July\or August\or September\or October\or November\or December\fi
2717     \space\number\day, \number\year
2718   \else\if 西暦
2719     \ifnum1=\iftdir\ifmdir0\else1\fi\else0\fi \kansuji\year
2720   \else\number\year\nobreak\fi 年
2721   \else
2722     \pltx@today@year
2723   \fi
2724   \ifnum1=\iftdir\ifmdir0\else1\fi\else0\fi
2725     \kansuji\month 月
2726     \kansuji\day 日
2727   \else
2728     \number\month\nobreak 月
2729     \number\day\nobreak 日
2730   \fi\fi}

```

■ハイフネーション例外  $\mathrm{T}_{\mathrm{E}}\mathrm{X}$  のハイフネーションルールの補足です（ペンディング：  
eng-lish）

```

2731 \hyphenation{ado-be post-script ghost-script phe-nom-e-no-log-i-cal man-u-script}

```

■ページ設定 ページ設定の初期化です。

```

2732 <article>\if@slide \pagestyle{empty} \else \pagestyle{plain} \fi
2733 <book>\if@report \pagestyle{plain} \else \pagestyle{headings} \fi
2734 <report|kiyou>\pagestyle{plain}
2735 <jspf>\pagestyle{headings}
2736 \pagenumbering{arabic}
2737 \if@twocolumn

```

```

2738 \twocolumn
2739 \sloppy
2740 \flushbottom
2741 \else
2742 \onecolumn
2743 \raggedbottom
2744 \fi
2745 \if@slide
2746 \renewcommand\kanjifamilydefault{\gtdefault}
2747 \renewcommand\familydefault{\sfdefault}
2748 \raggedright
2749 \xkanjiskip=0.1em\relax
2750 \fi

```

## 15 実験的コード

[2016-11-29] コミュニティ版 pL<sup>A</sup>T<sub>E</sub>X で新設されたテスト用パッケージ (exppl2e パッケージ) が文書クラスより先に読み込まれていた場合は, jsclasses もテスト版として動作します。この処置は jsarticle, jsbook, jsreport にのみ行い, jspf と kiyou は除外しておきます。exppl2e パッケージが読みこまれていない場合は通常版として動作しますので, ここで終了します。

```

2751 ⟨*article | book | report⟩
2752 \@ifpackageloaded{exppl2e}{\jsc@needspace@tchtrue}{\jsc@needspace@tchfalse}
2753 \ifjsc@needspace@tch\else
2754 \expandafter\endinput
2755 \fi

```

以下は実験的コードです。具体的には, 2016/11/29 の exppl2e パッケージで説明されている \@gnewline のパッチを入れてあります。

```

\@gnewline
2756 \def\@gnewline #1{%
2757   \ifvmode
2758     \@nolnerr
2759   \else
2760     \unskip \reserved@a {\reserved@a#1}\nobreak \hfil \break \hspace \z@
2761     \ignorespaces
2762   \fi}
2763 ⟨/article | book | report⟩
2764 ⟨/class⟩

```

以上です。