

# 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

2021/06/28

## 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 </class>
17 (*class | minijs)
18 \newif\ifjsc@needsp@tch
19 \jsc@needsp@tchfalse
20 </class | minijs>
21 (*class)

```

## 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。  
22 `\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。  
23 `\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。  
24 `\(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。  
25 `\(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。  
26 `\(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。  
27 `\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載しては 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， $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$  の `b5paper` は  $250\text{mm} \times 176\text{mm}$  ですが， $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$  の `b5paper` は  $257\text{mm} \times 182\text{mm}$  になっています。ここでは  $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$  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。

```

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

```

```

42 \setlength\paperwidth {257mm}}
43 \DeclareOption{b5paper}{%
44 \setlength\paperheight {257mm}%
45 \setlength\paperwidth {182mm}}
46 \DeclareOption{b6paper}{%
47 \setlength\paperheight {182mm}%
48 \setlength\paperwidth {128mm}}
49 \DeclareOption{a4j}{%
50 \setlength\paperheight {297mm}%
51 \setlength\paperwidth {210mm}}
52 \DeclareOption{a5j}{%
53 \setlength\paperheight {210mm}%
54 \setlength\paperwidth {148mm}}
55 \DeclareOption{b4j}{%
56 \setlength\paperheight {364mm}%
57 \setlength\paperwidth {257mm}}
58 \DeclareOption{b5j}{%
59 \setlength\paperheight {257mm}%
60 \setlength\paperwidth {182mm}}
61 \DeclareOption{a4var}{%
62 \setlength\paperheight {283mm}%
63 \setlength\paperwidth {210mm}}
64 \DeclareOption{b5var}{%
65 \setlength\paperheight {230mm}%
66 \setlength\paperwidth {182mm}}
67 \DeclareOption{letterpaper}{%
68 \setlength\paperheight {11in}%
69 \setlength\paperwidth {8.5in}}
70 \DeclareOption{legalpaper}{%
71 \setlength\paperheight {14in}%
72 \setlength\paperwidth {8.5in}}
73 \DeclareOption{executivepaper}{%
74 \setlength\paperheight {10.5in}%
75 \setlength\paperwidth {7.25in}}

```

■横置き 用紙の縦と横の長さを入れ換えます。

```

76 \newif\if@landscape
77 \@landscapefalse
78 \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ため、単純のため article のみで使えるオプションとしました。

```

79 \newif\if@slide
80 \@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。

```
81 \newcommand{\@ptsize}{0}
82 \newif\ifjsc@mag\jsc@magtrue
83 \newif\ifjsc@mag@xreal\jsc@mag@xrealfalse
84 \def\jsc@magscale{1}
85 <*article>
86 \DeclareOption{slide}{%
87   \@slidetrue\def\jsc@magscale{3.583}
88   \renewcommand{\@ptsize}{26}
89   \@landscapetrue\@titlepagetrue}
90 </article>
91 \DeclareOption{8pt}{\def\jsc@magscale{0.833}\renewcommand{\@ptsize}{-2}}
92 \DeclareOption{9pt}{\def\jsc@magscale{0.913}\renewcommand{\@ptsize}{-1}}
93 \DeclareOption{10pt}{\def\jsc@magscale{1}\renewcommand{\@ptsize}{0}}
94 \DeclareOption{11pt}{\def\jsc@magscale{1.095}\renewcommand{\@ptsize}{1}}
95 \DeclareOption{12pt}{\def\jsc@magscale{1.200}\renewcommand{\@ptsize}{2}}
96 \DeclareOption{14pt}{\def\jsc@magscale{1.440}\renewcommand{\@ptsize}{4}}
97 \DeclareOption{17pt}{\def\jsc@magscale{1.728}\renewcommand{\@ptsize}{7}}
98 \DeclareOption{20pt}{\def\jsc@magscale{2}\renewcommand{\@ptsize}{10}}
99 \DeclareOption{21pt}{\def\jsc@magscale{2.074}\renewcommand{\@ptsize}{11}}
100 \DeclareOption{25pt}{\def\jsc@magscale{2.488}\renewcommand{\@ptsize}{15}}
101 \DeclareOption{30pt}{\def\jsc@magscale{2.986}\renewcommand{\@ptsize}{20}}
102 \DeclareOption{36pt}{\def\jsc@magscale{3.583}\renewcommand{\@ptsize}{26}}
103 \DeclareOption{43pt}{\def\jsc@magscale{4.300}\renewcommand{\@ptsize}{33}}
104 \DeclareOption{12Q}{\def\jsc@magscale{0.923}\renewcommand{\@ptsize}{1200}}
105 \DeclareOption{14Q}{\def\jsc@magscale{1.077}\renewcommand{\@ptsize}{1400}}
106 \DeclareOption{10ptj}{\def\jsc@magscale{1.085}\renewcommand{\@ptsize}{1001}}
107 \DeclareOption{10.5ptj}{\def\jsc@magscale{1.139}\renewcommand{\@ptsize}{1051}}
108 \DeclareOption{11ptj}{\def\jsc@magscale{1.194}\renewcommand{\@ptsize}{1101}}
109 \DeclareOption{12ptj}{\def\jsc@magscale{1.302}\renewcommand{\@ptsize}{1201}}
110 \DeclareOption{usemag}{\jsc@magtrue\jsc@mag@xrealfalse}
111 \DeclareOption{nomag}{\jsc@magfalse\jsc@mag@xrealfalse}
112 \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  $\text{p}\text{L}\text{A}\text{T}\text{E}\text{X} 2_{\epsilon}$  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  $\text{p}\text{L}\text{A}\text{T}\text{E}\text{X} 2_{\epsilon}$  本体で宣言されています。

```

113 \hour\time \divide\hour by 60\relax
114 \@tempcnta\hour \multiply\@tempcnta 60\relax
115 \minute\time \advance\minute-\@tempcnta
116 \DeclareOption{tombow}{%
117   \tombowtrue \tombowdatetrue
118   \setlength{\@tombowwidth}{.1\p@}%
119   \@bannertoken{%
120     \jobname\space(\number\year-\two@digits\month-\two@digits\day
121     \space\two@digits\hour:\two@digits\minute)}}%
122   \maketombowbox}
123 \DeclareOption{tombo}{%
124   \tombowtrue \tombowdatefalse
125   \setlength{\@tombowwidth}{.1\p@}%
126   \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。

```

127 \DeclareOption{mentuke}{%
128   \tombowtrue \tombowdatefalse
129   \setlength{\@tombowwidth}{\z@}%
130   \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。

```

131 \DeclareOption{oneside}{\@twosidefalse \@mparswitchfalse}
132 \DeclareOption{twoside}{\@twosidetrue \@mparswitchtrue}
133 \DeclareOption{vartwoside}{\@twosidetrue \@mparswitchfalse}

```

■二段組 `twocolumn` で二段組になります。

```

134 \DeclareOption{onecolumn}{\@twocolumnfalse}
135 \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。

```

136 \DeclareOption{titlepage}{\@titlepagetrue}
137 \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 L<sup>A</sup>T<sub>E</sub>X 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。

```

138 (book|report)\DeclareOption{openright}{\@openrighttrue\@openleftfalse}

```

```

139 <book | report> \DeclareOption{openleft}{\@openlefttrue\@openrightfalse}
140 <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。

```

141 \def\eqnarray{%
142   \stepcounter{equation}%
143   \def\@currentlabel{\p@equation\theequation}%
144   \global\@eqnswtrue
145   \m@th
146   \global\@eqcnt\z@
147   \tabskip\@centering
148   \let\\\@eqnocr
149   $$\everycr{}\halign to\displaywidth\bgroup
150     \hskip\@centering$\displaystyle\tabskip\z@skip{##}$\@eqnse1
151     &\global\@eqcnt\@ne \hfil$\displaystyle{ }##{ }$\hfil
152     &\global\@eqcnt\tw@ $\displaystyle{##}$\hfil\tabskip\@centering
153     &\global\@eqcnt\thr@@ \hbxt@\z@\bgroup\hss##\egroup
154     \tabskip\z@skip
155     \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。

```

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

```

```

179     $\displaystyle{##}$\hfil \tabskip\@centering
180     &\global\@eqcnt\thr@@ \hbext@\z@\bgroup\hss##\egroup
181     \tabskip\z@skip\cr
182     }}

```

■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。

```

183 % \DeclareOption{openbib}{%
184 %   \AtEndOfPackage{%
185 %     \renewcommand\@openbib@code{%
186 %       \advance\leftmargin\bibindent
187 %       \itemindent -\bibindent
188 %       \listparindent \itemindent
189 %       \parsep \z@}%
190 %     \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。

```

191 \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。

```

192 \DeclareOption{draft}{\setlength\overfullrule{5pt}}
193 \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, dviptdpmx など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。

```

194 \newif\ifmingoth
195 \mingothfalse
196 \newif\ifjisfont
197 \jisfontfalse
198 \newif\if@jsc@uplatex

```



```

199 \@jsc@uplatexfalse
200 \newif\if@jsc@autodetect
201 \@jsc@autodetectfalse
202 \DeclareOption{winjis}{%
203   \ClassWarningNoLine{\jsc@clsname}{%
204     The option ‘winjis’ has been removed;\MessageBreak
205     Use ‘\string\usepackage{winjis}’ instead}}
206 \DeclareOption{mingoth}{\mingothtrue}
207 \DeclareOption{jis}{\jisfonttrue}
208 \DeclareOption{uplatex}{\@jsc@uplatextrue}
209 \DeclareOption{autodetect-engine}{\@jsc@autodetecttrue}
210 \def\jsc@JYn{\if@jsc@uplatex JY2\else JY1\fi}
211 \def\jsc@JTn{\if@jsc@uplatex JT2\else JT1\fi}
212 \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。

```

213 \newif\ifpapersize
214 \papersizefalse
215 \DeclareOption{papersize}{\papersizetrue}

```

■英語化 オプション english を新設しました。

```

216 \newif\if@english
217 \@englishfalse
218 \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。どちらでもお好きな方を使ってください。

```

219 (*book)
220 \newif\if@report
221 \@reportfalse
222 \DeclareOption{report}{\@reporttrue\@openrightfalse\@twosidefalse\@mparswitchfalse}
223 \end{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。

```

224 \newif\if@jslogo \@jslogotrue
225 \DeclareOption{jslogo}{\@jslogotrue}
226 \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。

```

227 (article)\ExecuteOptions{a4paper,oneside,onecolumn,notitlepage,final}
228 (book)\ExecuteOptions{a4paper,twoside,onecolumn,titlepage,openright,final}

```

```

229 <report>\ExecuteOptions{a4paper,oneside,onecolumn,titlepage,openany,final}
230 <jspf>\ExecuteOptions{a4paper,twoside,twocolumn,notitlepage,fleqn,final}
231 <kiyou>\ExecuteOptions{a4paper,twoside,twocolumn,notitlepage,final}
232 \ProcessOptions

```

後処理

```

233 \if@slide
234 \def\maybeblue{\ifundefined{ver@color.sty}{\color{blue}}{}}
235 \fi
236 \if@landscape
237 \setlength\@tempdima {\paperheight}
238 \setlength\paperheight{\paperwidth}
239 \setlength\paperwidth {\@tempdima}
240 \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] pL<sup>A</sup>T<sub>E</sub>X/ upL<sup>A</sup>T<sub>E</sub>X を自動判別するオプション `autodetect-engine` を新設しました。upL<sup>A</sup>T<sub>E</sub>X 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なります。

```

241 \ifnum \ifx\ucs\@undefined\z@\else\ucs"3000 \fi ="3000
242 \if@jsc@autodetect
243 \ClassInfo\jsc@clsname{Autodetected engine: upLaTeX}
244 \@jsc@uplatextrue
245 \g@addto@macro\@classoptionslist{,uplatex}
246 \fi
247 \if@jsc@uplatex\else
248 \ClassError\jsc@clsname
249 {You are running upLaTeX.\MessageBreak
250 Please use pLaTeX instead, or add 'uplatex' to\MessageBreak
251 the class option list}
252 {\@ehc}
253 \@jsc@uplatextrue
254 \fi

```

[2016-11-11] pL<sup>A</sup>T<sub>E</sub>X 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。

```

255 \else
256 \if@jsc@uplatex
257 \ClassError\jsc@clsname
258 {You are running pLaTeX.\MessageBreak
259 Please use upLaTeX instead, or remove 'uplatex' from\MessageBreak
260 the class option list}
261 {\@ehc}
262 \@jsc@uplatexfalse
263 \fi

```

```

264 \if@jsc@autodetect
265 \ClassInfo\jsc@clsname{Autodetected engine: pLaTeX}
266 \@jsc@uplatexfalse
267 \fi
268 \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 dvi<sub>pdfmx</sub> や最近の dvi<sub>out</sub> 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となります。

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 p<sub>La</sub>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

```
-O -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<sub>A</sub>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)。

```

269 \iftombow
270 \newdimen\stockwidth \newdimen\stockheight
271 \setlength{\stockwidth}{\paperwidth}
272 \setlength{\stockheight}{\paperheight}
273 \advance \stockwidth 2in
274 \advance \stockheight 2in
275 \fi
276 \ifpapersize
277 \iftombow
278 \edef\jsc@papersize@special{papersize=\the\stockwidth,\the\stockheight}
279 \else
280 \edef\jsc@papersize@special{papersize=\the\paperwidth,\the\paperheight}
281 \fi
282 \AtBeginDvi{\special{\jsc@papersize@special}}
283 \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。

```
284 (article | book | report)\if@slide\def\n@baseline{13}\else\def\n@baseline{16}\fi
285 (jspf)\def\n@baseline{14.554375}
286 (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@mpt` および `\jsc@mmm` 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。

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

```

318 \fi
319 \ifnum\@ptsize=10
320 \mag 2000
321 \def\inv@mag{0.5}
322 \fi
323 \ifnum\@ptsize=11
324 \mag 2074
325 \def\inv@mag{0.48216}
326 \fi
327 \ifnum\@ptsize=15
328 \mag 2488
329 \def\inv@mag{0.401929}
330 \fi
331 \ifnum\@ptsize=20
332 \mag 2986
333 \def\inv@mag{0.334896}
334 \fi
335 \ifnum\@ptsize=26
336 \mag 3583
337 \def\inv@mag{0.279096}
338 \fi
339 \ifnum\@ptsize=33
340 \mag 4300
341 \def\inv@mag{0.232558}
342 \fi
343 \ifnum\@ptsize=1200
344 \mag 923
345 \def\inv@mag{1.0834236}
346 \fi
347 \ifnum\@ptsize=1400
348 \mag 1077
349 \def\inv@mag{0.928505}
350 \fi
351 \ifnum\@ptsize=1001
352 \mag 1085
353 \def\inv@mag{0.921659}
354 \fi
355 \ifnum\@ptsize=1051
356 \mag 1139
357 \def\inv@mag{0.877963}
358 \fi
359 \ifnum\@ptsize=1101
360 \mag 1194
361 \def\inv@mag{0.837521}
362 \fi
363 \ifnum\@ptsize=1201
364 \mag 1302
365 \def\inv@mag{0.768049}
366 \fi

```

```

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

```

```

416 \jsc@get@external@font}
417 \def\jsc@fstrunc#1{%
418 \edef\jsc@tmpa{\strip@pt#1}%
419 \expandafter\jsc@fstrunc@a\jsc@tmpa.***\@nil}
420 \def\jsc@fstrunc@a#1.#2#3#4#5#6\@nil{%
421 \if#5*\else
422 \edef\jsc@tmpa{#1%
423 \ifnum#2#3>\z@ .#2\ifnum#3>\z@ #3\fi\fi}%
424 \fi}
425 \def\jsc@preadjust@extract@font{%
426 \let\jsc@req@size\f@size
427 \dimen@f@size\p@ \jsc@invscale\dimen@\jsc@magscale
428 \advance\dimen@.005pt\relax \jsc@fstrunc\dimen@
429 \let\jsc@ref@size\jsc@tmpa
430 \let\f@size\jsc@ref@size}
431 \def\execute@size@function#1{%
432 \let\jsc@cref@size\f@size
433 \let\f@size\jsc@req@size
434 \csname s@fct@#1\endcsname}
435 \let\jsc@DeclareErrorFont\DeclareErrorFont
436 \def\DeclareErrorFont#1#2#3#4#5{%
437 \@tempdimc#5\p@ \@tempdimc\jsc@magscale\@tempdimc
438 \edef\jsc@tmpa{{#1}{#2}{#3}{#4}{\strip@pt\@tempdimc}}
439 \expandafter\jsc@DeclareErrorFont\jsc@tmpa}
440 \def\gen@sfcnt{%
441 \edef\mandatory@arg{\mandatory@arg\jsc@cref@size}%
442 \empty@sfcnt}
443 \def\genb@sfcnt{%
444 \edef\mandatory@arg{%
445 \mandatory@arg\expandafter\genb@x\jsc@cref@size..\@@}%
446 \empty@sfcnt}
447 \DeclareErrorFont{OT1}{cmr}{m}{n}{10}
448 \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 449 \def\jsc@smallskip{\vspace\jsc@smallskipamount}
\jsc@bigskip 450 %\def\jsc@medskip{\vspace\jsc@medskipamount}
451 %\def\jsc@bigskip{\vspace\jsc@bigskipamount}

```

```

\jsc@smallskipamount
\jsc@medskipamount 452 \newskip\jsc@smallskipamount
\jsc@bigskipamount 453 \jsc@smallskipamount=3\jsc@mpt plus 1\jsc@mpt minus 1\jsc@mpt
454 %\newskip\jsc@medskipamount
455 %\jsc@medskipamount =6\jsc@mpt plus 2\jsc@mpt minus 2\jsc@mpt
456 %\newskip\jsc@bigskipamount

```

```
457 %\jsc@bigskipamoun =12\jsc@empt plus 4\jsc@empt minus 4\jsc@empt
```

`\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] トンボオプ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。

```
458 \setlength\paperwidth{\inv@mag\paperwidth}%
459 \setlength\paperheight{\inv@mag\paperheight}%
460 \iftombow
461   \setlength\stockwidth{\inv@mag\stockwidth}%
462   \setlength\stockheight{\inv@mag\stockheight}%
463 \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。

```
464 % \ifpapersize
465 %   \setlength{\@tempdima}{\paperwidth}
466 %   \setlength{\@tempdimb}{\paperheight}
467 %   \iftombow
468 %     \advance \@tempdima 2truein
469 %     \advance \@tempdimb 2truein
470 %   \fi
471 %   \AtBeginDvi{\special{pdf: pagesize width \the\@tempdima\space height \the\@tempdimb}}
472 % \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,  $\text{T}_{\text{E}}\text{X}$  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,  $\text{pT}_{\text{E}}\text{X}$  (アスキーが日本語化した  $\text{T}_{\text{E}}\text{X}$ ) 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 `\Cjascale` 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。

```

473 </class>
474 <*minijs>
475 %% min/goth -> jis/jisg (for pLaTeX only)
476 \ifnum\jis"2121="3000 \else
477 \@for\@tempa:=5,6,7,8,9,10,10.95,12,14.4,17.28,20.74,24.88\do{%
478   \expandafter\let\csname JY1/mc/m/n/\@tempa\endcsname\relax
479   \expandafter\let\csname JY1/gt/m/n/\@tempa\endcsname\relax
480   \expandafter\let\csname JT1/mc/m/n/\@tempa\endcsname\relax
481   \expandafter\let\csname JT1/gt/m/n/\@tempa\endcsname\relax
482 }
483 \def\Cjascale{0.924690}
484 \DeclareFontShape{JY1}{mc}{m}{n}{<-> s * [0.961] jis}{-}
485 \DeclareFontShape{JY1}{gt}{m}{n}{<-> s * [0.961] jisg}{-}
486 \DeclareFontShape{JT1}{mc}{m}{n}{<-> s * [0.961] tmin10}{-}
487 \DeclareFontShape{JT1}{gt}{m}{n}{<-> s * [0.961] tgoth10}{-}
488 \fi
489 </minijs>
490 <*class>
491 <*jspf>
492 \def\Cjascale{0.924690}
493 \ifmingoth
494   \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.961] \jsc@pfx@ min10}{-}
495   \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.961] \jsc@pfx@ goth10}{-}
496   \DeclareFontShape{\jsc@JTn}{mc}{m}{n}{<-> s * [0.961] \jsc@pfx@ tmin10}{-}
497   \DeclareFontShape{\jsc@JTn}{gt}{m}{n}{<-> s * [0.961] \jsc@pfx@ tgoth10}{-}
498 \else
499   \ifjisfont
500     \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.961] \jsc@pfx@ jis}{-}
501     \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.961] \jsc@pfx@ jisg}{-}
502     \DeclareFontShape{\jsc@JTn}{mc}{m}{n}{<-> s * [0.961] \jsc@pfx@ tmin10}{-}
503     \DeclareFontShape{\jsc@JTn}{gt}{m}{n}{<-> s * [0.961] \jsc@pfx@ tgoth10}{-}
504   \else
505     \if@jsc@uplatex
506       \DeclareFontShape{JY2}{mc}{m}{n}{<-> s * [0.924690] upjisir-h}{-}
507       \DeclareFontShape{JY2}{gt}{m}{n}{<-> s * [0.924690] upjisg-h}{-}
508       \DeclareFontShape{JT2}{mc}{m}{n}{<-> s * [0.924690] upjisir-v}{-}
509       \DeclareFontShape{JT2}{gt}{m}{n}{<-> s * [0.924690] upjisg-v}{-}
510     \else
511       \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.961] \jsc@pfx@ jis}{-}
512       \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.961] \jsc@pfx@ jisg}{-}
513       \DeclareFontShape{\jsc@JTn}{mc}{m}{n}{<-> s * [0.961] \jsc@pfx@ tmin10}{-}
514       \DeclareFontShape{\jsc@JTn}{gt}{m}{n}{<-> s * [0.961] \jsc@pfx@ tgoth10}{-}
515     \fi
516   \fi

```

```

517 \fi
518 \!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、 $9/(9.62216 * 72/72.27) = 0.93885$  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。
519 *jspf)
520 \def\Cjascale{0.903375}
521 \ifmingoth
522 \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ min10}{}
523 \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ goth10}{}
524 \DeclareFontShape{\jsc@JTn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tmin10}{}
525 \DeclareFontShape{\jsc@JTn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tgoth10}{}
526 \else
527 \ifjisfont
528 \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ jis}{}
529 \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ jisg}{}
530 \DeclareFontShape{\jsc@JTn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tmin10}{}
531 \DeclareFontShape{\jsc@JTn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tgoth10}{}
532 \else
533 \if@jsc@uplatex
534 \DeclareFontShape{JY2}{mc}{m}{n}{<-> s * [0.903375] upjisr-h}{}
535 \DeclareFontShape{JY2}{gt}{m}{n}{<-> s * [0.903375] upjisg-h}{}
536 \DeclareFontShape{JT2}{mc}{m}{n}{<-> s * [0.903375] upjisr-v}{}
537 \DeclareFontShape{JT2}{gt}{m}{n}{<-> s * [0.903375] upjisg-v}{}
538 \else
539 \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ jis}{}
540 \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ jisg}{}
541 \DeclareFontShape{\jsc@JTn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tmin10}{}
542 \DeclareFontShape{\jsc@JTn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tgoth10}{}
543 \fi
544 \fi
545 \fi
546 \!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。

```

547 % \DeclareFontShape{\jsc@JYn}{mc}{bx}{n}{<->ssub*gt/m/n}{} % in \jsc@JYnmc
548 % \DeclareFontShape{\jsc@JYn}{gt}{bx}{n}{<->ssub*gt/m/n}{} % in \jsc@JYngt
549 \DeclareFontShape{\jsc@JYn}{mc}{m}{it}{<->ssub*mc/m/n}{}

```

```

550 \DeclareFontShape{\jsc@JYn}{mc}{m}{sl}{<->ssub*mc/m/n}{-}
551 \DeclareFontShape{\jsc@JYn}{mc}{m}{sc}{<->ssub*mc/m/n}{-}
552 \DeclareFontShape{\jsc@JYn}{gt}{m}{it}{<->ssub*gt/m/n}{-}
553 \DeclareFontShape{\jsc@JYn}{gt}{m}{sl}{<->ssub*gt/m/n}{-}
554 \DeclareFontShape{\jsc@JYn}{mc}{bx}{it}{<->ssub*gt/m/n}{-}
555 \DeclareFontShape{\jsc@JYn}{mc}{bx}{sl}{<->ssub*gt/m/n}{-}
556 % \DeclareFontShape{\jsc@JTn}{mc}{bx}{n}{<->ssub*gt/m/n}{-} % in \jsc@JTnmc
557 % \DeclareFontShape{\jsc@JTn}{gt}{bx}{n}{<->ssub*gt/m/n}{-} % in \jsc@JTngt
558 \DeclareFontShape{\jsc@JTn}{mc}{m}{it}{<->ssub*mc/m/n}{-}
559 \DeclareFontShape{\jsc@JTn}{mc}{m}{sl}{<->ssub*mc/m/n}{-}
560 \DeclareFontShape{\jsc@JTn}{mc}{m}{sc}{<->ssub*mc/m/n}{-}
561 \DeclareFontShape{\jsc@JTn}{gt}{m}{it}{<->ssub*gt/m/n}{-}
562 \DeclareFontShape{\jsc@JTn}{gt}{m}{sl}{<->ssub*gt/m/n}{-}
563 \DeclareFontShape{\jsc@JTn}{mc}{bx}{it}{<->ssub*gt/m/n}{-}
564 \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。

```

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

```

```

591 \fi}
592 \else % 2020-02-02
593 \g@addto@macro\@rmfamilyhook
594 {\prepare@family@series@update@kanji{mc}\mcdefault}
595 \g@addto@macro\@sffamilyhook
596 {\prepare@family@series@update@kanji{gt}\gtdefault}
597 \g@addto@macro\@ttfamilyhook
598 {\prepare@family@series@update@kanji{gt}\gtdefault}
599 \fi
600 \else % --- for 2020-02-02 or older END & for 2020-10-01 BEGIN
601 \AddToHook{rmfamily}%
602 {\prepare@family@series@update@kanji{mc}\mcdefault}
603 \AddToHook{sffamily}%
604 {\prepare@family@series@update@kanji{gt}\gtdefault}
605 \AddToHook{ttfamily}%
606 {\prepare@family@series@update@kanji{gt}\gtdefault}
607 \fi % --- for 2020-10-01 END
608 </class | minijs>
609 <*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  
`\textgt` 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、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。

```

610 \ifx\DeclareFixJFMCJKTextFontCommand\undefined
611 \DeclareRobustCommand\textmc[1]{%
612   \relax\ifmmode \expandafter\nfss@text \fi{\mcfamily #1}}
613 \DeclareRobustCommand\textgt[1]{%
614   \relax\ifmmode \expandafter\nfss@text \fi{\gtfamily #1}}
615 \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。

```
616 \def\reDeclareMathAlphabet#1#2#3{%
617   \edef\@tempa{\expandafter\@gobble\string#2}%
618   \edef\@tempb{\expandafter\@gobble\string#3}%
619   \edef\@tempc{\string @\expandafter\@gobbletwo\string#2}%
620   \ifx\@tempc\@tempa%
621     \edef\@tempa{\expandafter\@gobbletwo\string#2}%
622     \edef\@tempb{\expandafter\@gobbletwo\string#3}%
623   \fi
624   \begingroup
625     \let\protect\noexpand
626     \def\@tempaa{\relax}%
627     \expandafter\ifx\csname RDMAorg@\@tempa\endcsname\relax
628       \edef\@tempaa{\expandafter\def\expandafter\noexpand%
629         \csname RDMAorg@\@tempa\endcsname{%
630           \expandafter\noexpand\csname\@tempa\endcsname}}%
631     \fi
632     \def\@tempbb{\relax}%
633     \expandafter\ifx\csname RDMAorg@\@tempb\endcsname\relax
634       \edef\@tempbb{\expandafter\def\expandafter\noexpand%
635         \csname RDMAorg@\@tempb\endcsname{%
636           \expandafter\noexpand\csname\@tempb\endcsname}}%
637     \fi
638     \edef\@tempc{\@tempaa\@tempbb}%
639     \expandafter\endgroup\@tempc%
640     \edef#1{\noexpand\protect\expandafter\noexpand\csname%
641       \expandafter\@gobble\string#1\space\space\endcsname}%
642     \expandafter\edef\csname\expandafter\@gobble\string#1\space\space\endcsname%
643       {\noexpand\DualLang@mathalph@bet%
644         {\expandafter\noexpand\csname RDMAorg@\@tempa\endcsname}}%
645       {\expandafter\noexpand\csname RDMAorg@\@tempb\endcsname}}%
646   }%
647 }
648 \@onlypreamble\reDeclareMathAlphabet
649 \def\DualLang@mathalph@bet#1#2{%
650   \relax\ifmmode
651     \ifx\math@bgroup\bgroup%      2e normal style   (\mathrm{...})
652       \bgroup\let\DualLang@Mfontsw\DLMfontsw@standard
653     \else
654       \ifx\math@bgroup\relax%     2e two letter style (\rm->\mathrm)
655         \let\DualLang@Mfontsw\DLMfontsw@oldstyle
656       \else
657         \ifx\math@bgroup\@empty%  2.09 oldfont style ({\mathrm ...})
658           \let\DualLang@Mfontsw\DLMfontsw@oldfont
659         \else%
660           \panic! assume 2e normal style
661         \bgroup\let\DualLang@Mfontsw\DLMfontsw@standard
662       \fi
663     \fi
664   }
```

```

662     \fi
663     \fi
664     \else
665     \let\DualLang@Mfontsw\@firstoftwo
666     \fi
667     \DualLang@Mfontsw{#1}{#2}%
668 }
669 \def\DLMfontsw@standard#1#2#3{#1{#2{#3}}\egroup}
670 \def\DLMfontsw@oldstyle#1#2{#1\relax\@fontswitch\relax{#2}}
671 \def\DLMfontsw@oldfont#1#2{#1\relax#2\relax}
672 \if@enablejfam
673   \DeclareSymbolFont{mincho}{\jsc@JYn}{mc}{m}{n}
674   \DeclareSymbolFontAlphabet{\mathmc}{mincho}
675   \SetSymbolFont{mincho}{bold}{\jsc@JYn}{gt}{m}{n}
676   \jfam\symmincho
677   \DeclareMathAlphabet{\mathgt}{\jsc@JYn}{gt}{m}{n}
678   \AtBeginDocument{%
679     \reDeclareMathAlphabet{\mathrm}{\@mathrm}{\@mathmc}
680     \reDeclareMathAlphabet{\mathbf}{\@mathbf}{\@mathgt}}
681 \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。

```
682 % \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。

```

683 \prebreakpenalty\jis"2147=10000      % 5000   '
684 \postbreakpenalty\jis"2148=10000     % 5000   "
685 \prebreakpenalty\jis"2149=10000     % 5000   "

```

「 $\TeX$  !」「 $\pi$ 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。

```

686 \inhibitxspcode'!=1
687 \inhibitxspcode'\pi=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

```
688 % \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。

```
689 \xspcode'+=3
690 \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。

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

732 \xspcode{^^a9=3  
733 \xspcode{^^aa=3  
734 \xspcode{^^ab=3  
735 \xspcode{^^ac=3  
736 \xspcode{^^ad=3  
737 \xspcode{^^ae=3  
738 \xspcode{^^af=3  
739 \xspcode{^^b0=3  
740 \xspcode{^^b1=3  
741 \xspcode{^^b2=3  
742 \xspcode{^^b3=3  
743 \xspcode{^^b4=3  
744 \xspcode{^^b5=3  
745 \xspcode{^^b6=3  
746 \xspcode{^^b7=3  
747 \xspcode{^^b8=3  
748 \xspcode{^^b9=3  
749 \xspcode{^^ba=3  
750 \xspcode{^^bb=3  
751 \xspcode{^^bc=3  
752 \xspcode{^^bd=3  
753 \xspcode{^^be=3  
754 \xspcode{^^bf=3  
755 \xspcode{^^c0=3  
756 \xspcode{^^c1=3  
757 \xspcode{^^c2=3  
758 \xspcode{^^c3=3  
759 \xspcode{^^c4=3  
760 \xspcode{^^c5=3  
761 \xspcode{^^c6=3  
762 \xspcode{^^c7=3  
763 \xspcode{^^c8=3  
764 \xspcode{^^c9=3  
765 \xspcode{^^ca=3  
766 \xspcode{^^cb=3  
767 \xspcode{^^cc=3  
768 \xspcode{^^cd=3  
769 \xspcode{^^ce=3  
770 \xspcode{^^cf=3  
771 \xspcode{^^d0=3  
772 \xspcode{^^d1=3  
773 \xspcode{^^d2=3  
774 \xspcode{^^d3=3  
775 \xspcode{^^d4=3  
776 \xspcode{^^d5=3  
777 \xspcode{^^d6=3  
778 \xspcode{^^d7=3  
779 \xspcode{^^d8=3  
780 \xspcode{^^d9=3



```

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

```

\@ 欧文といえば、 $\text{\LaTeX}$  の  $\text{\def\@{\spacefactor\@m}}$  という定義 ( $\@m$  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。

```

819 \def\@{\spacefactor3000{}}

```

## 4 フォントサイズ

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

```
\setfontsize{\normalsize}{10}{16}
```

のようにして行います。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。

```
820 </class>
821 <*class | minijs>
822 %% \setfontsize with \parindent and \(\x)kanjiskip settings
823 \def\setfontsize#1#2#3{%
824 <minijs> \@nomath#1%
825 \ifx\protect\@typeset@protect
826 \let\@currsize#1%
827 \fi
828 \fontsize{#2}{#3}\selectfont
829 \ifdim\parindent>\z@
830 <class> \if@english
831 <class> \parindent=1em
832 <class> \else
```

```

833     \parindent=1zw
834 <class>   \fi
835   \fi
836   \kanjiskip=0zw plus .1zw minus .01zw
837 <class>   \ifdim\xkanjiskip>\z@
838 <class>   \if@slide \xkanjiskip=0.1em \else
839     \xkanjiskip=0.25em plus 0.15em minus 0.06em
840 <class>   \fi
841 <class>   \fi
842 }
843 </class | minijs>
844 <*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。

```

845 \def\jsc@setfontsize#1#2#3{%
846   \@setfontsize#1{#2\jsc@empt}{#3\jsc@empt}}

```

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。

```

847 \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

`\widebaselines` `\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、しばば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。

```

848 \newif\ifnarrowbaselines
849 \if@english
850   \narrowbaselinestrue
851 \fi
852 \def\narrowbaselines{%
853   \narrowbaselinestrue
854   \skip0=\abovedisplayskip
855   \skip2=\abovedisplayshortskip
856   \skip4=\belowdisplayskip
857   \skip6=\belowdisplayshortskip
858   \@currsize\selectfont
859   \abovedisplayskip=\skip0
860   \abovedisplayshortskip=\skip2
861   \belowdisplayskip=\skip4
862   \belowdisplayshortskip=\skip6\relax}
863 \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 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。

```
864 \renewcommand{\normalsize}{%
865   \ifnarrowbaselines
866     \jsc@setfontsize\normalsize\@xpt\@xipt
867   \else
868     \jsc@setfontsize\normalsize\@xpt{\n@baseline}%
869   \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、微調節してやることにしました。

```
870 \abovedisplayskip 11\jsc@mpt \@plus3\jsc@mpt \@minus4\jsc@mpt
871 \abovedisplayshortskip \z@ \@plus3\jsc@mpt
872 \belowdisplayskip 9\jsc@mpt \@plus3\jsc@mpt \@minus4\jsc@mpt
873 \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。

```
874 \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。

```
875 </class>
876 <*class | minijs>
877 %% initialize
878 \normalsize
879 </class | minijs>
880 <*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 (`lzw`) 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。

```
881 \setbox0\hbox{\char\jis"3441}%
882 \setlength\Cht{\ht0}
883 \setlength\Cdp{\dp0}
884 \setlength\Cwd{\wd0}
885 \setlength\Cvs{\baselineskip}
886 \setlength\Chs{\wd0}
887 \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。

```

888 \newcommand{\small}{%
889   \ifnarrowbaselines
890 (!kiyou)   \jsc@setfontsize\small\@ixpt{11}%
891 (kiyou)    \jsc@setfontsize\small{8.8888}{11}%
892   \else
893 (!kiyou)   \jsc@setfontsize\small\@ixpt{13}%
894 (kiyou)    \jsc@setfontsize\small{8.8888}{13.2418}%
895   \fi
896 \abovedisplayskip 9\jsc@empt \@plus3\jsc@empt \@minus4\jsc@empt
897 \abovedisplayshortskip \z@ \@plus3\jsc@empt
898 \belowdisplayskip \abovedisplayskip
899 \belowdisplayshortskip \belowdisplayskip
900 \def\@listi{\leftmargin\leftmargini
901           \topsep \z@
902           \parsep \z@
903           \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。

```

904 \newcommand{\footnotesize}{%
905   \ifnarrowbaselines
906 (!kiyou)   \jsc@setfontsize\footnotesize\@viipt{9.5}%
907 (kiyou)    \jsc@setfontsize\footnotesize{8.8888}{11}%
908   \else
909 (!kiyou)   \jsc@setfontsize\footnotesize\@viipt{11}%
910 (kiyou)    \jsc@setfontsize\footnotesize{8.8888}{13.2418}%
911   \fi
912 \abovedisplayskip 6\jsc@empt \@plus2\jsc@empt \@minus3\jsc@empt
913 \abovedisplayshortskip \z@ \@plus2\jsc@empt
914 \belowdisplayskip \abovedisplayskip
915 \belowdisplayshortskip \belowdisplayskip
916 \def\@listi{\leftmargin\leftmargini
917           \topsep \z@
918           \parsep \z@
919           \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 920 \newcommand{\scriptsize}{\jsc@setfontsize\scriptsize\@viipt\@viipt}
\huge 921 \newcommand{\tiny}{\jsc@setfontsize\tiny\@vpt\@vipt}
\Huge
\HUGE

```

```

922 \if@twocolumn
923 (!kiyou) \newcommand{\large}{\jsc@setfontsize\large\xiipt{\n@baseline}}
924 (kiyou) \newcommand{\large}{\jsc@setfontsize\large{11.111}{\n@baseline}}
925 \else
926 (!kiyou) \newcommand{\large}{\jsc@setfontsize\large\xiipt{17}}
927 (kiyou) \newcommand{\large}{\jsc@setfontsize\large{11.111}{17}}
928 \fi
929 (!kiyou)\newcommand{\Large}{\jsc@setfontsize\Large\xivpt{21}}
930 (kiyou)\newcommand{\Large}{\jsc@setfontsize\Large{12.222}{21}}
931 \newcommand{\LARGE}{\jsc@setfontsize\LARGE\xviipt{25}}
932 \newcommand{\huge}{\jsc@setfontsize\huge\@xxpt{28}}
933 \newcommand{\Huge}{\jsc@setfontsize\Huge\@xxvpt{33}}
934 \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。

```

935 \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。『`PLATEX2ε` 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。

```

936 % \newcommand{\headfont}{\bfseries}
937 \newcommand{\headfont}{\gtfamily\sffamily}
938 % \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 でしたが、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。

```

939 (!kiyou)\setlength\columnsep{2zw}
940 (kiyou)\setlength\columnsep{28truebp}
941 \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 942 \setlength\lineskip{1\jsc@mp}
\normallineskiplimit 943 \setlength\normallineskip{1\jsc@mp}
944 \setlength\lineskiplimit{1\jsc@mp}
945 \setlength\normallineskiplimit{1\jsc@mp}

```

`\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。

```

946 \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。

```

947 \setlength\parskip{\z@}
948 \if@slide
949 \setlength\parindent{0zw}
950 \else
951 \setlength\parindent{1zw}
952 \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

`\@medpenalty` になっています。ここはオリジナル通りです。

```

\@highpenalty 953 \@lowpenalty 51
954 \@medpenalty 151
955 \@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。

```

956 % \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。

```

957 % \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 ですが、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。

```
958 \setlength\topskip{1.38zw}%% from 10\jsc@mpt (2016-08-17)
959 \if@slide
960   \setlength\headheight{0\jsc@mpt}
961 \else
962   \setlength\headheight{20\jsc@mpt}%% from 2\topskip (2016-08-17); from \topskip (2003-06-26)
963 \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。

```
964 (*article | kiyou)
965 \if@slide
966   \setlength\footskip{0pt}
967 \else
968   \setlength\footskip{0.03367\paperheight}
969   \ifdim\footskip<\baselineskip
970     \setlength\footskip{\baselineskip}
971   \fi
972 \fi
973 </article | kiyou>
974 (jspf)\setlength\footskip{9\jsc@mmm}
975 (*book)
976 \if@report
977   \setlength\footskip{0.03367\paperheight}
978   \ifdim\footskip<\baselineskip
979     \setlength\footskip{\baselineskip}
980   \fi
981 \else
982   \setlength\footskip{0pt}
983 \fi
984 </book>
985 (*report)
986 \setlength\footskip{0.03367\paperheight}
987 \ifdim\footskip<\baselineskip
988   \setlength\footskip{\baselineskip}
989 \fi
990 </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)。

```
991 <*article>
992 \if@slide
993   \setlength\headsep{0\jsc@empt}
994   \addtolength\headsep{-\topskip}%% added (2016-10-08)
995   \addtolength\headsep{10\jsc@empt}%% added (2016-10-08)
996 \else
997   \setlength\headsep{\footskip}
998   \addtolength\headsep{-\topskip}
999 \fi
1000 </article>
1001 <*book>
1002 \if@report
1003   \setlength\headsep{\footskip}
1004   \addtolength\headsep{-\topskip}
1005 \else
1006   \setlength\headsep{6\jsc@mmm}
1007   \addtolength\headsep{-\topskip}%% added (2016-10-08)
1008   \addtolength\headsep{10\jsc@empt}%% added (2016-10-08)
1009 \fi
1010 </book>
1011 <*report>
1012 \setlength\headsep{\footskip}
1013 \addtolength\headsep{-\topskip}
1014 </report>
1015 <*jspf>
1016 \setlength\headsep{9\jsc@mmm}
1017 \addtolength\headsep{-\topskip}
1018 </jspf>
1019 <*kiyou>
1020 \setlength\headheight{0\jsc@empt}
1021 \setlength\headsep{0\jsc@empt}
1022 \addtolength\headsep{-\topskip}%% added (2016-10-08)
1023 \addtolength\headsep{10\jsc@empt}%% added (2016-10-08)
1024 </kiyou>
```

`\maxdepth` `\maxdepth` は本文最下行の最大の深さで, plain  $\text{T}_\text{E}\text{X}$  や  $\text{L}^{\text{A}}\text{T}_\text{E}\text{X}$  2.09 では 4pt に固定でした。  $\text{L}^{\text{A}}\text{T}_\text{E}\text{X}2\text{e}$  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。

```
1025 \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。

```
1026 \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。

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

```

1064 \setlength\textwidth{\fullwidth}
1065 \jspf
1066 *kiyou)
1067 \setlength\fullwidth{48zw}
1068 \addtolength\fullwidth{\columnsep}
1069 \setlength\textwidth{\fullwidth}
1070 \jkiyou)

```

`\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)。

```

1071 (*article | book | report)
1072 \if@slide
1073 \setlength{\textheight}{0.95\paperheight}
1074 \else
1075 \setlength{\textheight}{0.83\paperheight}
1076 \fi
1077 \addtolength{\textheight}{-10\jsc@mpt}%% from -\topskip (2016-10-08); from -\headheight (2003-
1078 \addtolength{\textheight}{-\headsep}
1079 \addtolength{\textheight}{-\footskip}
1080 \addtolength{\textheight}{-\topskip}
1081 \divide\textheight\baselineskip
1082 \multiply\textheight\baselineskip
1083 \jarticle | book | report)
1084 \jspf)\setlength{\textheight}{51\baselineskip}
1085 \kiyou)\setlength{\textheight}{47\baselineskip}
1086 \addtolength{\textheight}{\topskip}
1087 \addtolength{\textheight}{0.1\jsc@mpt}
1088 \jspf)\setlength{\mathindent}{10\jsc@mmm}

```

`\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 L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> 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。

```
1089 \def\flushbottom{%
1090   \def\@textbottom{\vskip \z@ \@plus.1\jcs@mpt}%
1091   \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` みどうしの最小の間隔です。

```
1092 \setlength\marginparsep{\columnsep}
1093 \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。

```
1094 \setlength{\oddsidemargin}{\paperwidth}
1095 \addtolength{\oddsidemargin}{-\fullwidth}
1096 \setlength{\oddsidemargin}{.5\oddsidemargin}
1097 \iftombow
1098   \addtolength{\oddsidemargin}{-1in}
1099 \else
1100   \addtolength{\oddsidemargin}{-\inv@mag in}
1101 \fi
1102 \setlength{\evensidemargin}{\oddsidemargin}
1103 \if@mparswitch
1104   \addtolength{\evensidemargin}{\fullwidth}
1105   \addtolength{\evensidemargin}{-\textwidth}
1106 \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。

```
1107 \setlength\marginparwidth{\paperwidth}
1108 \addtolength\marginparwidth{-\oddsidemargin}
1109 \addtolength\marginparwidth{-\inv@mag in}
1110 \addtolength\marginparwidth{-\textwidth}
1111 \addtolength\marginparwidth{-10\jcs@mmm}
1112 \addtolength\marginparwidth{-\marginparsep}
1113 \@tempdima=1zw
1114 \divide\marginparwidth\@tempdima
1115 \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)。

```
1116 \setlength\topmargin{\paperheight}
1117 \addtolength\topmargin{-\textheight}
1118 \if@slide
1119   \addtolength\topmargin{-\headheight}
1120 \else
1121   \addtolength\topmargin{-10\jsc@empt}%% from -\topskip (2016-10-08); from -\headheight (2003-0
1122 \fi
1123 \addtolength\topmargin{-\headsep}
1124 \addtolength\topmargin{-\footskip}
1125 \setlength\topmargin{0.5\topmargin}
1126 (kiyou)\setlength\topmargin{81truebp}
1127 \iftombow
1128   \addtolength\topmargin{-1in}
1129 \else
1130   \addtolength\topmargin{-\inv@mag in}
1131 \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。

```
1132 {\footnotesize\global\setlength\footnotesep{\baselineskip}}
1133 \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。

```
1134 \setlength{\skip\footins}{16\jsc@empt \@plus 5\jsc@empt \@minus 2\jsc@empt}
```

■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。

```
1135 \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。

```
1136 \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。

1137 `\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。

1138 `\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。

1139 `\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。

1140 `\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。

1141 `\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。

1142 `\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。

1143 `\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。

1144 `\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文との距離です。

`\intextsep`

1145 `\setlength\floatsep {12\jsc@empt \@plus 2\jsc@empt \@minus 2\jsc@empt}`

1146 `\setlength\textfloatsep{20\jsc@empt \@plus 2\jsc@empt \@minus 4\jsc@empt}`

1147 `\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` 1148 `\setlength\dblfloatsep {12\jsc@empt \@plus 2\jsc@empt \@minus 2\jsc@empt}`

1149 `\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` 1150 `\setlength\@fptop{0\jsc@empt \@plus 1fil}`

1151 `\setlength\@fpsep{8\jsc@empt \@plus 2fil}`

1152 `\setlength\@fpbot{0\jsc@empt \@plus 1fil}`

`\@dblfpptop` 段抜きフロートについての値です。

`\@dblfpsep` 1153 `\setlength\@dblfpptop{0\jsc@empt \@plus 1fil}`

`\@dblfpbot` 1154 `\setlength\@dblfpsep{8\jsc@empt \@plus 2fil}`

1155 `\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 pL<sup>A</sup>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` 1. `\pltx@cleartorightpage` : 右ページになるまでページを繰る命令
- `\pltx@cleartoevenpage` 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。

```
1156 (*article | book | report)
1157 \def\pltx@cleartorightpage{\clearpage\if@twoside
1158   \ifodd\c@page
1159     \iftdir
1160       \hbox{}\thispagestyle{empty}\newpage
1161       \if@twocolumn\hbox{}\newpage\fi
1162     \fi
1163   \else
1164     \ifydir
1165       \hbox{}\thispagestyle{empty}\newpage
1166       \if@twocolumn\hbox{}\newpage\fi
1167     \fi
1168   \fi\fi}
1169 \def\pltx@cleartoleftpage{\clearpage\if@twoside
1170   \ifodd\c@page
1171     \ifydir
1172       \hbox{}\thispagestyle{empty}\newpage
1173       \if@twocolumn\hbox{}\newpage\fi
1174     \fi
1175   \else
1176     \iftdir
1177       \hbox{}\thispagestyle{empty}\newpage
1178       \if@twocolumn\hbox{}\newpage\fi
1179     \fi
1180   \fi\fi}
1181 \def\pltx@cleartooddpage{\clearpage\if@twoside
1182   \ifodd\c@page\else
1183     \hbox{}\thispagestyle{empty}\newpage
1184     \if@twocolumn\hbox{}\newpage\fi
1185   \fi\fi}
1186 \def\pltx@cleartoevenpage{\clearpage\if@twoside
1187   \ifodd\c@page
1188     \hbox{}\thispagestyle{empty}\newpage
1189     \if@twocolumn\hbox{}\newpage\fi
1190   \fi\fi}
```

```
1191 </article | book | report>
```

`\cleardoublepage` [2017-02-24] コミュニティ版 p $\text{\LaTeX}$  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。

```
1192 (*book | report)
```

```
1193 \if@openleft
```

```
1194 \let\cleardoublepage\pltx@cleartoleftpage
```

```
1195 \else\if@openright
```

```
1196 \let\cleardoublepage\pltx@cleartorightpage
```

```
1197 \fi\fi
```

```
1198 </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` で変更する場合がこれにあたります。しかし、同一ページに複数の `\markboth` があると、おかしな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。

```
1199 % \def\ps@empty{%
```

```
1200 % \let\@mkboth\@gobbletwo
```

```
1201 % \let\@oddhead\@empty
```

```
1202 % \let\@oddfoot\@empty
```



```

1203 % \let\@evenhead\@empty
1204 % \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。

1205 \def\ps@plainfoot{%
1206 \let\@mkboth\@gobbletwo
1207 \let\@oddhead\@empty
1208 \def\@oddfoot{\normalfont\hfil\thepage\hfil}%
1209 \let\@evenhead\@empty
1210 \let\@evenfoot\@oddfoot}
1211 \def\ps@plainhead{%
1212 \let\@mkboth\@gobbletwo
1213 \let\@oddfoot\@empty
1214 \let\@evenfoot\@empty
1215 \def\@evenhead{%
1216 \if@mparswitch \hss \fi
1217 \hbox to \fullwidth{\textbf{\thepage}\hfil}%
1218 \if@mparswitch\else \hss \fi}%
1219 \def\@oddhead{%
1220 \hbox to \fullwidth{\hfil\textbf{\thepage}}\hss}}
1221 (book)\if@report \let\ps@plain\ps@plainfoot \else \let\ps@plain\ps@plainhead \fi
1222 (!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。

```

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

```

```

1243 \def\ps@headings{%
1244 \let\@oddfoot\@empty
1245 \def\@oddhead{%
1246 \underline{%
1247 \hbox to \fullwidth{\rightmark}\hfil\textbf{\thepage}}\hss}%
1248 \let\@mkboth\markboth
1249 \def\sectionmark##1{\markright{%
1250 \ifnum \c@secnumdepth >\z@ \thesection \hskip1zw\fi
1251 ##1}}%
1252 \fi
1253 </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）。

```

1254 (*book | report)
1255 \newif\if@omit@number
1256 \def\ps@headings{%
1257 \let\@oddfoot\@empty
1258 \let\@evenfoot\@empty
1259 \def\@evenhead{%
1260 \ifmparswitch \hss \fi
1261 \underline{\hbox to \fullwidth{\autoxspacing
1262 \textbf{\thepage}\hfil\leftmark}}}%
1263 \ifmparswitch\else \hss \fi}%
1264 \def\@oddhead{\underline{\hbox to \fullwidth{\autoxspacing
1265 {\if@twoside\rightmark\else\leftmark\fi}\hfil\textbf{\thepage}}}\hss}%
1266 \let\@mkboth\markboth
1267 \def\chaptermark##1{\markboth{%
1268 \ifnum \c@secnumdepth >\m@ne
1269 (book) \if@mainmatter
1270 \if@omit@number\else
1271 \@chapapp\thechapter\@chappos\hskip1zw
1272 \fi
1273 (book) \fi
1274 \fi
1275 ##1}{}}%
1276 \def\sectionmark##1{\markright{%
1277 \ifnum \c@secnumdepth >\z@ \thesection \hskip1zw\fi
1278 ##1}}%
1279 </book | report)

```

最後は学会誌の場合です。

```

1280 (*jspf)
1281 \def\ps@headings{%
1282 \def\@oddfoot{\normalfont\hfil\thepage\hfil}
1283 \def\@evenfoot{\normalfont\hfil\thepage\hfil}
1284 \def\@oddhead{\normalfont\hfil \@title \hfil}
1285 \def\@evenhead{\normalfont\hfil プラズマ・核融合学会誌 \hfil}}
1286 </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。

```
1287 \def\ps@myheadings{%
1288   \let\@oddfont\@empty\let\@evenfont\@empty
1289   \def\@evenhead{%
1290     \if@mparswitch \hss \fi%
1291     \hbox to \fullwidth{\thepage\hfil\leftmark}%
1292     \if@mparswitch\else \hss \fi}%
1293   \def\@oddhead{%
1294     \hbox to \fullwidth{\rightmark\hfil\thepage}\hss}%
1295   \let\@mkboth\@gobbletwo
1296 (book|report) \let\chaptermark\@gobble
1297 \let\sectionmark\@gobble
1298 (!book&!report) \let\subsectionmark\@gobble
1299 }
```

## 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 1300 % \newcommand*{\title}[1]{\gdef\@title{#1}}
1301 % \newcommand*{\author}[1]{\gdef\@author{#1}}
1302 % \newcommand*{\date}[1]{\gdef\@date{#1}}
1303 % \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。

```
\eauthor 1304 <*jspf>
\keywords 1305 \newcommand*{\etitle}[1]{\gdef\@etitle{#1}}
1306 \newcommand*{\eauthor}[1]{\gdef\@eauthor{#1}}
1307 \newcommand*{\keywords}[1]{\gdef\@keywords{#1}}
1308 \newcommand*{\email}[1]{\gdef\authors@mail{#1}}
1309 \newcommand*{\AuthorsEmail}[1]{\gdef\authors@mail{author's e-mail:\ #1}}
1310 </jspf>
```

`\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。

```
1311 \def\plainifnotempty{%
1312   \ifx \@oddhead \@empty
1313     \ifx \@oddfont \@empty
1314     \else
```

```

1315     \thispagestyle{plainfoot}%
1316     \fi
1317     \else
1318     \thispagestyle{plainhead}%
1319     \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。

```

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

```

```

1357 \end{titlepage}%
1358 \setcounter{footnote}{0}%
1359 \global\let\thanks\relax
1360 \global\let\maketitle\relax
1361 \global\let\@thanks\@empty
1362 \global\let\@author\@empty
1363 \global\let\@date\@empty
1364 \global\let\@title\@empty
1365 \global\let\title\relax
1366 \global\let\author\relax
1367 \global\let\date\relax
1368 \global\let\and\relax
1369 }%
1370 \else
1371 \newcommand{\maketitle}{\par
1372 \begingroup
1373 \renewcommand\thefootnote{\@fnsymbol\c@footnote}%
1374 \def\@makefnmark{\rlap{\@textsuperscript{\normalfont\@thefnmark}}}%
1375 \long\def\@makefntext##1{\advance\leftskip 3zw
1376 \parindent 1zw\noindent
1377 \llap{\@textsuperscript{\normalfont\@thefnmark}\hskip0.3zw}##1}%
1378 \if@twocolumn
1379 \ifnum \col@number=\@ne
1380 \maketitle
1381 \else
1382 \twocolumn[\maketitle]%
1383 \fi
1384 \else
1385 \newpage
1386 \global\@topnum\z@ % Prevents figures from going at top of page.
1387 \maketitle
1388 \fi
1389 \plainifnotempty
1390 \@thanks
1391 \endgroup
1392 \setcounter{footnote}{0}%
1393 \global\let\thanks\relax
1394 \global\let\maketitle\relax
1395 \global\let\@thanks\@empty
1396 \global\let\@author\@empty
1397 \global\let\@date\@empty
1398 \global\let\@title\@empty
1399 \global\let\title\relax
1400 \global\let\author\relax
1401 \global\let\date\relax
1402 \global\let\and\relax
1403 }

```

`\@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。

```
1404 \def\@maketitle{%
1405   \newpage\null
1406   \vskip 2em
1407   \begin{center}%
1408     \let\footnote\thanks
1409     {\LARGE \@title \par}%
1410     \vskip 1.5em
1411     {\large
1412       \lineskip .5em
1413       \begin{tabular}[t]{c}%
1414         \@author
1415         \end{tabular}\par}%
1416     \vskip 1em
1417     {\large \@date}%
1418   \end{center}%
1419   \par\vskip 1.5em
1420 (article | report | kiyou) \ifvoid\@abstractbox\else\centerline{\box\@abstractbox}\vskip1.5em\fi
1421 }
1422 \fi
1423 </article | book | report | kiyou>
1424 <*jspf>
1425 \newcommand{\maketitle}{\par
1426   \begingroup
1427     \renewcommand\thefootnote{\@fnsymbol\c@footnote}%
1428     \def\@makefnmark{\rlap{\@textsuperscript{\normalfont\@thefnmark}}}%
1429     \long\def\@makefntext##1{\advance\leftskip 3zw
1430       \parindent 1zw\noindent
1431       \llap{\@textsuperscript{\normalfont\@thefnmark}\hskip0.3zw}##1}%
1432     \twocolumn[\@maketitle]%
1433     \plainifnotempty
1434     \@thanks
1435   \endgroup
1436   \setcounter{footnote}{0}%
1437   \global\let\thanks\relax
1438   \global\let\maketitle\relax
1439   \global\let\@thanks\@empty
1440   \global\let\@author\@empty
1441   \global\let\@date\@empty
1442 % \global\let\@title\@empty % \@title は柱に使う
1443   \global\let\title\relax
1444   \global\let\author\relax
1445   \global\let\date\relax
1446   \global\let\and\relax
1447   \ifx\authors@mail\@undefined\else{%
1448     \def\@makefntext{\advance\leftskip 3zw \parindent -3zw}%
1449     \footnotetext[0]{\itshape\authors@mail}%
1450   }\fi
```

```

1451 \global\let\authors@mail\undefined}
1452 \def\@maketitle{%
1453 \newpage\null
1454 \vskip 6em % used to be 2em
1455 \begin{center}
1456 \let\footnote\thanks
1457 \ifx\@title\undefined\else{\LARGE\headfont\@title\par}\fi
1458 \lineskip .5em
1459 \ifx\@author\undefined\else
1460 \vskip 1em
1461 \begin{tabular}[t]{c}%
1462 \@author
1463 \end{tabular}\par
1464 \fi
1465 \ifx\@etitle\undefined\else
1466 \vskip 1em
1467 {\large \@etitle \par}%
1468 \fi
1469 \ifx\@eauthor\undefined\else
1470 \vskip 1em
1471 \begin{tabular}[t]{c}%
1472 \@eauthor
1473 \end{tabular}\par
1474 \fi
1475 \vskip 1em
1476 \@date
1477 \end{center}
1478 \vskip 1.5em
1479 \centerline{\box\@abstractbox}
1480 \ifx\@keywords\undefined\else
1481 \vskip 1.5em
1482 \centerline{\parbox{157\jsc@mmm}{\textsf{Keywords:}}\ \small\@keywords}}
1483 \fi
1484 \vskip 1.5em}
1485 \</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。

```
1486 \def\@startsection#1#2#3#4#5#6{%
1487   \if@noskipsec \leavevmode \fi
1488   \par
1489 % 見出し上の空きを \@tempskipa にセットする
1490   \@tempskipa #4\relax
1491 % \@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
1492   \if@english \@afterindentfalse \else \@afterindenttrue \fi
1493 % 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
1494   \ifdim \@tempskipa <\z@
1495     \@tempskipa -\@tempskipa \@afterindentfalse
1496   \fi
1497   \if@nobreak
1498 %   \everypar{\everyparhook}% これは間違い
1499     \everypar{}%
1500   \else
1501     \addpenalty\@secpenalty
1502 % 次の行は削除
1503 %   \addvspace\@tempskipa
1504 % 次の \noindent まで追加
1505     \ifdim \@tempskipa >\z@
1506       \if@slide\else
1507         \null
1508         \vspace*{-\baselineskip}%
1509       \fi
1510       \vskip\@tempskipa
1511     \fi
1512   \fi
1513   \noindent
1514 % 追加終わり
1515   \@ifstar
1516   {\@ssect{#3}{#4}{#5}{#6}}%
```



```

1517   {\@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。
1518 \def\@sect#1#2#3#4#5#6[#7]#8{%
1519   \ifnum #2>\c@secnumdepth
1520     \let\@svsec\@empty
1521   \else
1522     \refstepcounter{#1}%
1523     \protected@edef\@svsec{\@secntformat{#1}\relax}%
1524   \fi
1525 % 見出し後の空きを \@tempkipa にセット
1526   \@tempkipa #5\relax
1527 % 条件判断の順序を入れ替えました
1528   \ifdim \@tempkipa<\z@
1529     \def\@svsechd{%
1530       #6{\hskip #3\relax
1531         \@svsec #8}%
1532       \csname #1mark\endcsname{#7}%
1533       \addcontentsline{toc}{#1}{%
1534         \ifnum #2>\c@secnumdepth \else
1535           \protect\numberline{\csname the#1\endcsname}%
1536         \fi
1537         #7}}% 目次にフルネームを載せるなら #8
1538   \else
1539     \begingroup
1540     \interlinepenalty \@M % 下から移動
1541     #6{%
1542       \@hangfrom{\hskip #3\relax\@svsec}%
1543     % \interlinepenalty \@M % 上に移動
1544     #8\@@par}%
1545   \endgroup
1546   \csname #1mark\endcsname{#7}%
1547   \addcontentsline{toc}{#1}{%
1548     \ifnum #2>\c@secnumdepth \else
1549       \protect\numberline{\csname the#1\endcsname}%
1550     \fi
1551     #7}% 目次にフルネームを載せるならここは #8
1552   \fi
1553   \@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。

```

1554 \def\@xsect#1{%
1555 % 見出しの後ろの空きを \@tempkipa にセット
1556   \@tempkipa #1\relax
1557 % 条件判断の順序を変えました

```

```

1558 \ifdim \@tempskipa<\z@
1559   \nobreakfalse
1560   \global\@noskipsectrue
1561   \everypar{%
1562     \if@noskipsec
1563       \global\@noskipsecfalse
1564       {\setbox\z@\lastbox}%
1565       \clubpenalty\@M
1566       \begingroup \@svsechd \endgroup
1567       \unskip
1568       \@tempskipa #1\relax
1569       \hskip -\@tempskipa
1570     \else
1571       \clubpenalty \@clubpenalty
1572       \everypar{\everyparhook}%
1573     \fi\everyparhook}%
1574 \else
1575   \par \nobreak
1576   \vskip \@tempskipa
1577   \@afterheading
1578 \fi
1579 \if@slide
1580   {\vskip\if@twocolumn-5\jsc@empt\else-6\jsc@empt\fi
1581    \maybeblue\hrule height0\jsc@empt depth1\jsc@empt
1582    \vskip\if@twocolumn 4\jsc@empt\else 7\jsc@empt\fi\relax}%
1583 \fi
1584 \par % 2000-12-18
1585 \ignorespaces}
1586 \def\@ssect#1#2#3#4#5{%
1587   \@tempskipa #3\relax
1588   \ifdim \@tempskipa<\z@
1589     \def\@svsechd{#4{\hskip #1\relax #5}}%
1590   \else
1591     \begingroup
1592     #4{%
1593       \@hangfrom{\hskip #1}%
1594       \interlinepenalty \@M #5\@@par}%
1595     \endgroup
1596   \fi
1597   \@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 1598 \newcommand*\chaptermark[1]{}
1599 % \newcommand*\sectionmark[1]{}
\subsubsectionmark 1600 % \newcommand*\subsectionmark[1]{}
\paragraphmark 1601 % \newcommand*\subsubsectionmark[1]{}
\subparagraphmark

```

```
1602 % \newcommand*\paragraphmark}[1]{  
1603 % \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。

```
1604 (!book&!report)\setcounter{secnumdepth}{3}  
1605 (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 1606 \newcounter{part}  
1607 (book|report)\newcounter{chapter}  
\c@subsubsection 1608 (book|report)\newcounter{section}[chapter]  
1609 (!book&!report)\newcounter{section}  
\c@subparagraph 1610 \newcounter{subsection}[section]  
1611 \newcounter{subsubsection}[subsection]  
1612 \newcounter{paragraph}[subsubsection]  
1613 \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。

```
1614 \renewcommand{\thepart}{\@Roman\c@part}  
1615 (!book&!report)% \renewcommand{\thesection}{\@arabic\c@section}  
1616 (!book&!report)\renewcommand{\thesection}{\presectionname\@arabic\c@section\postsectionname}  
1617 (!book&!report)\renewcommand{\thesubsection}{\@arabic\c@section.\@arabic\c@subsection}  
1618 (*book|report)  
1619 \renewcommand{\thechapter}{\@arabic\c@chapter}  
1620 \renewcommand{\thesection}{\thechapter.\@arabic\c@section}  
1621 \renewcommand{\thesubsection}{\thesection.\@arabic\c@subsection}  
1622 
```

`\@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。

```
1629 <book | report> \newcommand{\@chapapp}{\prechaptername}
```

```
1630 <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した。 `openany` かどうかにかかわらず奇数ページまで繰るよう修正することで, 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)。

```
1631 <*book>
1632 \newcommand\frontmatter{%
1633   \pltx@cleartooddpage
1634   \@mainmatterfalse
1635   \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。

```
1636 \newcommand\mainmatter{%
1637   \pltx@cleartooddpage
1638   \@mainmattertrue
1639   \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。

```
1640 \newcommand\backmatter{%
1641   \if@openleft
1642     \cleardoublepage
1643   \else\if@openright
1644     \cleardoublepage
1645   \else
1646     \clearpage
1647   \fi\fi
1648   \@mainmatterfalse}
1649 </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。

```
1650 <!*book&!report)
1651 \newcommand\part{%
1652   \if@noskipsec \leavevmode \fi
1653   \par
1654   \addvspace{4ex}%
1655   \if@english \@afterindentfalse \else \@afterindenttrue \fi
1656   \secdef\@part\@spart}
1657 </!*book&!report)
```

`book` および `report` クラスの場合は、少し複雑です。

```
1658 <*book | report)
1659 \newcommand\part{%
1660   \if@openleft
1661     \cleardoublepage
1662   \else\if@openright
1663     \cleardoublepage
1664   \else
1665     \clearpage
1666   \fi\fi
1667   \thispagestyle{empty}% 欧文用標準スタイルでは plain
1668   \if@twocolumn
1669     \onecolumn
1670     \@restonecoltrue
1671   \else
1672     \@restonecolfalse
1673   \fi
1674   \null\vfil
1675   \secdef\@part\@spart}
1676 </*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。

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

```

1684 \addcontentsline{toc}{part}{#1}%
1685 \fi
1686 \markboth{}{}%
1687 {\parindent\z@
1688 \raggedright
1689 \interlinepenalty \@M
1690 \normalfont
1691 \ifnum \c@secnumdepth >\m@ne
1692 \Large\headfont\prepartname\thepart\postpartname
1693 \par\nobreak
1694 \fi
1695 \huge \headfont #2%
1696 \markboth{}{}\par}%
1697 \nobreak
1698 \vskip 3ex
1699 \@afterheading}
1700 </!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。

```

1701 (*book | report)
1702 \def\@part[#1]#2{%
1703 \ifnum \c@secnumdepth >-2\relax
1704 \refstepcounter{part}%
1705 \addcontentsline{toc}{part}{%
1706 \prepartname\thepart\postpartname\hspace{1zw}#1}%
1707 \else
1708 \addcontentsline{toc}{part}{#1}%
1709 \fi
1710 \markboth{}{}%
1711 {\centering
1712 \interlinepenalty \@M
1713 \normalfont
1714 \ifnum \c@secnumdepth >-2\relax
1715 \huge\headfont \prepartname\thepart\postpartname
1716 \par\vskip20\jsc@mpt
1717 \fi
1718 \Huge \headfont #2\par}%
1719 \@endpart}
1720 </book | report)

```

\@spart 番号を付けない部です。

```

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

```

```

1729 \afterheading}
1730 </!book&!report)
1731 (*book | report)
1732 \def\@spart#1{%
1733     \centering
1734     \interlinepenalty \@M
1735     \normalfont
1736     \Huge \headfont #1\par}%
1737 \@endpart}
1738 </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。

```

1739 (*book | report)
1740 \def\@endpart{\vfil\newpage
1741     \if@twoside
1742     \if@openleft %% added (2017/02/24)
1743     \null\thispagestyle{empty}\newpage
1744     \else\if@openright %% added (2016/12/13)
1745     \null\thispagestyle{empty}\newpage
1746     \fi\fi %% added (2016/12/13, 2017/02/24)
1747     \fi
1748     \if@restonecol
1749     \twocolumn
1750     \fi}
1751 </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。

```

1752 (*book | report)
1753 \newcommand{\chapter}{%
1754     \if@openleft\cleardoublepage\else
1755     \if@openright\cleardoublepage\else\clearpage\fi\fi
1756     \plainifnotempty % 元: \thispagestyle{plain}
1757     \global\@topnum\z@
1758     \if@english \@afterindentfalse \else \@afterindenttrue \fi
1759     \secdef
1760     {\@omit@numberfalse\@chapter}%
1761     {\@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。

```

1762 \def\@chapter[#1]#2{%
1763   \ifnum \c@secnumdepth >\m@ne
1764 (book)   \if@mainmatter
1765           \refstepcounter{chapter}%
1766           \typeout{\@chapapp\thechapter\@chappos}%
1767           \addcontentsline{toc}{chapter}%
1768             {\protect\numberline
1769              % {\if@english\thechapter\else\@chapapp\thechapter\@chappos\fi}%
1770              {\@chapapp\thechapter\@chappos}%
1771              #1}%
1772 (book)   \else\addcontentsline{toc}{chapter}{#1}\fi
1773   \else
1774     \addcontentsline{toc}{chapter}{#1}%
1775     \fi
1776     \chaptermark{#1}%
1777     \addtocontents{lof}{\protect\addvspace{10\jsc@empt}}%
1778     \addtocontents{lot}{\protect\addvspace{10\jsc@empt}}%
1779     \if@twocolumn
1780       \@topnewpage[\@makechapterhead{#2}]%
1781     \else
1782       \@makechapterhead{#2}%
1783       \@afterheading
1784     \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。

```

1785 \def\@makechapterhead#1{%
1786   \vspace*{2\Cvs}% 欧文は 50pt
1787   {\parindent \z@ \raggedright \normalfont
1788    \ifnum \c@secnumdepth >\m@ne
1789 (book)   \if@mainmatter
1790           \huge\headfont \@chapapp\thechapter\@chappos
1791           \par\nobreak
1792           \vskip \Cvs % 欧文は 20pt
1793 (book)   \fi
1794     \fi
1795     \interlinepenalty\M
1796     \Huge \headfont #1\par\nobreak
1797     \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。

```

1798 \def\@schapter#1{%
1799   \chaptermark{#1}%
1800   \if@twocolumn
1801     \@topnewpage[\@makeschapterhead{#1}]%
1802   \else
1803     \@makeschapterhead{#1}\@afterheading
1804   \fi}

```

\@makeschapterhead 番号なしの章見出しです。



```

1805 \def\@makeschapterhead#1{%
1806   \vspace*{2\Cvs}% 欧文は 50pt
1807   {\parindent \z@ \raggedright
1808     \normalfont
1809     \interlinepenalty\@M
1810     \Huge \headfont #1\par\nobreak
1811     \vskip 3\Cvs}} % 欧文は 40pt
1812 </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。

```

1813 \if@twocolumn
1814   \newcommand{\section}{%
1815     (jspf)\ifx\maketitle\relax\else\maketitle\fi
1816     \@startsection{section}{1}{\z@}%
1817     (!kiyou)   {0.6\Cvs}{0.4\Cvs}%
1818     (kiyou)    {\Cvs}{0.5\Cvs}%
1819     %   {\normalfont\large\headfont\@secapp}}
1820     {\normalfont\large\headfont\raggedright}}
1821 \else
1822   \newcommand{\section}{%
1823     \if@slide\clearpage\fi
1824     \@startsection{section}{1}{\z@}%
1825     {\Cvs \@plus.5\Cdp \@minus.2\Cdp}% 前アキ
1826     {.5\Cvs \@plus.3\Cdp}% 後アキ
1827     %   {\normalfont\Large\headfont\@secapp}}
1828     {\normalfont\Large\headfont\raggedright}}
1829 \fi

```

`\subsection` 同上です。

```

1830 \if@twocolumn
1831   \newcommand{\subsection}{\@startsection{subsection}{2}{\z@}%
1832     {\z@}{\if@slide .4\Cvs \else \z@ \fi}%
1833     {\normalfont\normalsize\headfont}}
1834 \else
1835   \newcommand{\subsection}{\@startsection{subsection}{2}{\z@}%
1836     {\Cvs \@plus.5\Cdp \@minus.2\Cdp}% 前アキ
1837     {.5\Cvs \@plus.3\Cdp}% 後アキ
1838     {\normalfont\large\headfont}}
1839 \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)。

```

1840 \if@twocolumn
1841   \newcommand{\subsubsection}{\@startsection{subsubsection}{3}{\z@}%

```

```

1842     {\z@}{\if@slide .4\Cvs \else \z@ \fi}%
1843     {\normalfont\normalsize\headfont}}
1844 \else
1845   \newcommand{\subsubsection}{\@startsection{subsubsection}{3}{\z@}%
1846     {\Cvs \@plus.5\Cdp \@minus.2\Cdp}%
1847     {\if@slide .5\Cvs \@plus.3\Cdp \else \z@ \fi}%
1848     {\normalfont\normalsize\headfont}}
1849 \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。

```

1850 (!jspf)\newcommand{\jsParagraphMark}{■}
1851 \if@twocolumn
1852   \newcommand{\paragraph}{\@startsection{paragraph}{4}{\z@}%
1853     {\z@}{\if@slide .4\Cvs \else -1zw\fi}% 改行せず 1zw のアキ
1854   (jspf)   {\normalfont\normalsize\headfont}}
1855 (!jspf)   {\normalfont\normalsize\headfont\jsParagraphMark}}
1856 \else
1857   \newcommand{\paragraph}{\@startsection{paragraph}{4}{\z@}%
1858     {0.5\Cvs \@plus.5\Cdp \@minus.2\Cdp}%
1859     {\if@slide .5\Cvs \@plus.3\Cdp \else -1zw\fi}% 改行せず 1zw のアキ
1860   (jspf)   {\normalfont\normalsize\headfont}}
1861 (!jspf)   {\normalfont\normalsize\headfont\jsParagraphMark}}
1862 \fi

```

`\subparagraph` 見出しの後ろで改行されません。

```

1863 \if@twocolumn
1864   \newcommand{\subparagraph}{\@startsection{subparagraph}{5}{\z@}%
1865     {\z@}{\if@slide .4\Cvs \@plus.3\Cdp \else -1zw\fi}%
1866     {\normalfont\normalsize\headfont}}
1867 \else
1868   \newcommand{\subparagraph}{\@startsection{subparagraph}{5}{\z@}%
1869     {\z@}{\if@slide .5\Cvs \@plus.3\Cdp \else -1zw\fi}%
1870     {\normalfont\normalsize\headfont}}
1871 \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。

```
1872 \if@slide
1873   \setlength\leftmargini{1zw}
1874 \else
1875   \if@twocolumn
1876     \setlength\leftmargini{2zw}
1877   \else
1878     \setlength\leftmargini{3zw}
1879   \fi
1880 \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 1881 \if@slide
\leftmarginv 1882   \setlength\leftmarginii {1zw}
1883   \setlength\leftmarginiii{1zw}
\leftmarginvi 1884   \setlength\leftmarginiv {1zw}
1885   \setlength\leftmarginv {1zw}
1886   \setlength\leftmarginvi {1zw}
1887 \else
1888   \setlength\leftmarginii {2zw}
1889   \setlength\leftmarginiii{2zw}
1890   \setlength\leftmarginiv {2zw}
1891   \setlength\leftmarginv {1zw}
1892   \setlength\leftmarginvi {1zw}
1893 \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。

```
1894 \setlength \labelsep {0.5zw} % .5em
1895 \setlength \labelwidth{\leftmargini}
1896 \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。

```
1897 \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 1898 \@beginparpenalty -\@lowpenalty
\@itempenalty 1899 \@endparpenalty -\@lowpenalty
1900 \@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  $\pm_{0.1}^{0.2}$  `\baselineskip` を思い切って外しました。

```
1901 \def\@listi{\leftmargin\leftmargini
1902   \parsep \z@
1903   \topsep 0.5\baselineskip
1904   \itemsep \z@ \relax}
1905 \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)。

```
1906 \@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 1907 \def\@listii{\leftmargin\leftmarginii
\@listiv 1908   \labelwidth\leftmarginii \advance\labelwidth-\labelsep
1909   \topsep \z@
\@listv 1910   \parsep \z@
\@listvi 1911   \itemsep\parsep}
1912 \def\@listiii{\leftmargin\leftmarginiii
1913   \labelwidth\leftmarginiii \advance\labelwidth-\labelsep
1914   \topsep \z@
1915   \parsep \z@
1916   \itemsep\parsep}
1917 \def\@listiv {\leftmargin\leftmarginiv
1918   \labelwidth\leftmarginiv
1919   \advance\labelwidth-\labelsep}
1920 \def\@listv {\leftmargin\leftmarginv
1921   \labelwidth\leftmarginv
1922   \advance\labelwidth-\labelsep}
1923 \def\@listvi {\leftmargin\leftmarginvi
1924   \labelwidth\leftmarginvi
1925   \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。

```
1926 \renewcommand{\theenumi}{\@arabic\c@enumi}
1927 \renewcommand{\theenumii}{\@alph\c@enumii}
1928 \renewcommand{\theenumiii}{\@roman\c@enumiii}
1929 \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  
`\labelenumii` 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。

```
\labelenumiv 1930 \newcommand{\labelenumi}{\theenumi.}
1931 \newcommand{\labelenumii}{\inhibitglue (\theenumii) \inhibitglue}
1932 \newcommand{\labelenumiii}{\theenumiii.}
1933 \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  
`\p@enumiii` 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 1934 \renewcommand{\p@enumii}{\theenumi}
1935 \renewcommand{\p@enumiii}{\theenumi\inhibitglue (\theenumii) }
1936 \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 1937 \newcommand\labelitemi{\textbullet}
\labelitemiii 1938 \newcommand\labelitemii{\normalfont\bfseries \textendash}
1939 \newcommand\labelitemiii{\textasteriskcentered}
\labelitemiv 1940 \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。

```
1941 \newenvironment{description}{%
1942 \list{}{%
1943 \labelwidth=\leftmargin
1944 \labelsep=1zw
1945 \advance \labelwidth by -\labelsep
1946 \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  
(たとえば `\hspace{1zw}`) を入れるのもいいと思います。

```
1947 \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。

```
1948 (*book)
1949 \newenvironment{abstract}{%
```

```

1950 \begin{list}{}{%
1951 \listparindent=1zw
1952 \itemindent=\listparindent
1953 \rightmargin=0pt
1954 \leftmargin=5zw}\item[]{}\end{list}\vspace{\baselineskip}}
1955 \end{book}
1956 (*article | report | kiyou)
1957 \newbox\@abstractbox
1958 \if@titlepage
1959 \newenvironment{abstract}{%
1960 \titlepage
1961 \null\vfil
1962 \@beginparpenalty\@lowpenalty
1963 \begin{center}%
1964 \headfont \abstractname
1965 \@endparpenalty\@M
1966 \end{center}}%
1967 {\par\vfil\null\endtitlepage}
1968 \else
1969 \newenvironment{abstract}{%
1970 \if@twocolumn
1971 \ifx\maketitle\relax
1972 \section*{\abstractname}%
1973 \else
1974 \global\setbox\@abstractbox\hbox\bgroup
1975 \begin{minipage}[b]{\textwidth}
1976 \small\parindent1zw
1977 \begin{center}%
1978 {\headfont \abstractname\vspace{-.5em}\vspace{\z@}}%
1979 \end{center}%
1980 \list{}{%
1981 \listparindent\parindent
1982 \itemindent \listparindent
1983 \rightmargin \leftmargin}%
1984 \item\relax
1985 \fi
1986 \else
1987 \small
1988 \begin{center}%
1989 {\headfont \abstractname\vspace{-.5em}\vspace{\z@}}%
1990 \end{center}%
1991 \list{}{%
1992 \listparindent\parindent
1993 \itemindent \listparindent
1994 \rightmargin \leftmargin}%
1995 \item\relax
1996 \fi}\if@twocolumn
1997 \ifx\maketitle\relax
1998 \else

```

```

1999      \endlist\end{minipage}\egroup
2000      \fi
2001      \else
2002      \endlist
2003      \fi}
2004 \fi
2005 </article | report | kiyou>
2006 <*jspf>
2007 \newbox\@abstractbox
2008 \newenvironment{abstract}{%
2009   \global\setbox\@abstractbox\hbox\bgroup
2010   \begin{minipage}[b]{157\jsc@mmm}{\sffamily Abstract}\par
2011     \small
2012     \if@english \parindent6\jsc@mmm \else \parindent1zw \fi}%
2013   {\end{minipage}\egroup}
2014 </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。

```

2015 <*jspf>
2016 %\newbox\@keywordsbox
2017 %\newenvironment{keywords}{%
2018 %   \global\setbox\@keywordsbox\hbox\bgroup
2019 %   \begin{minipage}[b]{1570\jsc@mmm}{\sffamily Keywords:}\par
2020 %     \small\parindent0zw}%
2021 %   {\end{minipage}\egroup}
2022 </jspf>

```

### ■verse 環境

**verse** 詩のための `verse` 環境です。

```

2023 \newenvironment{verse}{%
2024   \let \\\=@centercr
2025   \list{}{%
2026     \itemsep \z@
2027     \itemindent -2zw % 元: -1.5em
2028     \listparindent\itemindent
2029     \rightmargin \z@
2030     \advance\leftmargin 2zw}% 元: 1.5em
2031   \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。

```

2032 \newenvironment{quotation}{%
2033   \list{}{%

```

```

2034 \listparindent\parindent
2035 \itemindent\listparindent
2036 \rightmargin \z0}%
2037 \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。

```

2038 \newenvironment{quote}%
2039 {\list{}{\rightmargin\z0}\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。

```

2040 \def\@begintheorem#1#2{\trivlist\labelsep=1zw
2041 \item[\hskip \labelsep{\headfont #1\ #2}]}
2042 \def\@opargbegintheorem#1#2#3{\trivlist\labelsep=1zw
2043 \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、こちらで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。

```

2044 \newenvironment{titlepage}{%
2045 (book) \pltx@cleartooddpage %% 2017-02-24
2046 \if@twocolumn
2047 \@restonecoltrue\onecolumn
2048 \else
2049 \@restonecolfalse\newpage
2050 \fi
2051 \thispagestyle{empty}%
2052 \ifodd\c@page\setcounter{page}\@ne\else\setcounter{page}\z@\fi %% 2017-02-24
2053 }%
2054 {\if@restonecol\twocolumn \else \newpage \fi
2055 \if@twoside\else
2056 \setcounter{page}\@ne
2057 \fi}

```



## ■付録

`\appendix` 本文と付録を分離するコマンドです。

```
2058 (*!book&!report)
2059 \newcommand{\appendix}{\par
2060   \setcounter{section}{0}%
2061   \setcounter{subsection}{0}%
2062   \gdef\presectionname{\appendixname}%
2063   \gdef\postsectionname{}}
2064 % \gdef\thesection{\@Alph\c@section}% [2003-03-02]
2065 \gdef\thesection{\presectionname\@Alph\c@section\postsectionname}%
2066 \gdef\thesubsection{\@Alph\c@section.\@arabic\c@subsection}}
2067  $\!/book&!report)
2068 (*book | report)
2069 \newcommand{\appendix}{\par
2070   \setcounter{chapter}{0}%
2071   \setcounter{section}{0}%
2072   \gdef\@chapapp{\appendixname}%
2073   \gdef\@chappos{}}
2074 \gdef\thechapter{\@Alph\c@chapter}}
2075  $\!/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。

```
2076 \setlength\arraycolsep{5\jsc@mppt}
```

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

```
2077 \setlength\tabcolsep{6\jsc@mppt}
```

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

```
2078 \setlength\arrayrulewidth{.4\jsc@mppt}
```

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

```
2079 \setlength\doublerulesep{2\jsc@mppt}
```

### ■tabbing 環境

`\tabbingsep` \’ コマンドで入るアキです。

```
2080 \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。

```
2081 \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。

```
2082 \setlength\fboxsep{3\jsc@empt}
2083 \setlength\fboxrule{.4\jsc@empt}
```

### ■equation と eqnarray 環境

`\theequation` 数式番号を出力するコマンドです。

```
2084 (!book&!report)\renewcommand \theequation {\@arabic\c@equation}
2085 (*book | report)
2086 \@addtoreset{equation}{chapter}
2087 \renewcommand\theequation
2088 {\ifnum \c@chapter>\z@ \thechapter.\fi \@arabic\c@equation}
2089 </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。

```
2090 % \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。

```
2091 % \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。

```
2092 % \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。

```
2093 <!*book&!report>
2094 \newcounter{figure}
2095 \renewcommand \thefigure {\@arabic\c@figure}
2096 </!book&!report>
2097 <*book | report>
2098 \newcounter{figure}[chapter]
2099 \renewcommand \thefigure
2100     {\ifnum \c@chapter>\z@ \thechapter.\fi \@arabic\c@figure}
2101 </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、ここでは外しました。

```
\ext@figure 2102 \def\fps@figure{tbp}
2103 \def\ftype@figure{1}
\fnun@figure 2104 \def\ext@figure{lof}
2105 \def\fnun@figure{\figurename\nobreak\thefigure}
```

`figure` \* 形式は段抜きフロートです。

```
figure* 2106 \newenvironment{figure}%
2107     {\@float{figure}}%
2108     {\end@float}
2109 \newenvironment{figure*}%
2110     {\@dblfloat{figure}}%
2111     {\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。

```
2112 <!*book&!report>
2113 \newcounter{table}
2114 \renewcommand\thetable{\@arabic\c@table}
2115 </!book&!report>
2116 <*book | report>
2117 \newcounter{table}[chapter]
2118 \renewcommand \thetable
2119     {\ifnum \c@chapter>\z@ \thechapter.\fi \@arabic\c@table}
2120 </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、ここでは外しました。

```
\ext@table 2121 \def\fps@table{tbp}
2122 \def\ftype@table{2}
\fnun@table 2123 \def\ext@table{lot}
2124 \def\fnun@table{\tablename\nobreak\thetable}
```

`table` \* は段抜きフロートです。

`table*`

```

2125 \newenvironment{table}%
2126         {\@float{table}}%
2127         {\end@float}
2128 \newenvironment{table*}%
2129         {\@dblfloat{table}}%
2130         {\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。

```

2131 \newlength\abovecaptionskip
2132 \newlength\belowcaptionskip
2133 \setlength\abovecaptionskip{5\jsc@empt} % 元: 10\p@
2134 \setlength\belowcaptionskip{5\jsc@empt} % 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。

```

2135 (*!jspf)
2136 % \long\def\@makecaption#1#2{\small
2137 %   \advance\leftskip10\jsc@mmm
2138 %   \advance\rightskip10\jsc@mmm
2139 %   \vskip\abovecaptionskip
2140 %   \sbox\@tempboxa{#1\hskip1zw\relax #2}%
2141 %   \ifdim \wd\@tempboxa >\hsize
2142 %     #1\hskip1zw\relax #2\par
2143 %   \else
2144 %     \global \@minipagefalse
2145 %     \hb@xt@\hsize{\hfil\box\@tempboxa\hfil}%
2146 %   \fi
2147 %   \vskip\belowcaptionskip}}
2148 \long\def\@makecaption#1#2{\small
2149   \advance\leftskip .0628\linewidth
2150   \advance\rightskip .0628\linewidth
2151   \vskip\abovecaptionskip
2152   \sbox\@tempboxa{#1\hskip1zw\relax #2}%
2153   \ifdim \wd\@tempboxa <\hsize \centering \fi
2154   #1{\hskip1zw\relax}#2\par
2155   \vskip\belowcaptionskip}}

```

```

2156 </jpsf>
2157 <*jpsf>
2158 \long\def\@makecaption#1#2{%
2159   \vskip\abovecaptionskip
2160   \sbox\@tempboxa{\small\sffamily #1\quad #2}%
2161   \ifdim \wd\@tempboxa >\hsize
2162     {\small\sffamily
2163     \list{#1}{%
2164       \renewcommand{\makelabel}[1]{##1\hfil}
2165       \itemsep \z@
2166       \itemindent \z@
2167       \labelsep \z@
2168       \labelwidth 11\jsc@mmm
2169       \listparindent\z@
2170       \leftmargin 11\jsc@mmm}\item\relax #2\endlist}
2171   \else
2172     \global \@minipagefalse
2173     \hb@xt@\hsize{\hfil\box\@tempboxa\hfil}%
2174   \fi
2175   \vskip\belowcaptionskip}
2176 </jpsf>

```

## 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 2177 \DeclareOldFontCommand{\mc}{\normalfont\mcfamily}{\mathmc}
\rm 2178 \DeclareOldFontCommand{\gt}{\normalfont\gtfamily}{\mathgt}
\sf 2179 \DeclareOldFontCommand{\rm}{\normalfont\rmfamily}{\mathrm}
\tt 2180 \DeclareOldFontCommand{\sf}{\normalfont\sffamily}{\mathsf}
\tt 2181 \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すコマンドは `\mdseries` です。

```

2182 \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すコマンドは `\upshape` です。

```

2183 \DeclareOldFontCommand{\it}{\normalfont\itshape}{\mathit}
2184 \DeclareOldFontCommand{\sl}{\normalfont\slshape}{\@nomath\sl}
2185 \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`

```
2186 \DeclareRobustCommand*\cal{\@fontswitch\relax\mathcal}
2187 \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`  $\geq$  `\@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。

```
2188 \newcommand\@pnumwidth{1.55em}
2189 \newcommand\@tocrmarg{2.55em}
2190 \newcommand\@dotsep{4.5}
2191 (!book&!report)\setcounter{tocdepth}{2}
2192 (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)

```
2193 \newdimen\jsc@tocl@width
2194 \newcommand{\tableofcontents}{%
2195 (*book | report)
2196   \settowidth\jsc@tocl@width{\headfont\prechaptername\postchaptername}%
2197   \settowidth\@tempdima{\headfont\appendixname}%
2198   \ifdim\jsc@tocl@width<\@tempdima \setlength\jsc@tocl@width{\@tempdima}\fi
2199   \ifdim\jsc@tocl@width<2zw \divide\jsc@tocl@width by 2 \advance\jsc@tocl@width 1zw\fi
2200   \if@twocolumn
2201     \@restonecoltrue\onecolumn
2202   \else
2203     \@restonecolfalse
2204   \fi
2205   \chapter*{\contentsname}%
2206   \@mkboth{\contentsname}{}%
2207 </book | report)
2208 (*!book&!report)
2209   \settowidth\jsc@tocl@width{\headfont\presectionname\postsectionname}%
2210   \settowidth\@tempdima{\headfont\appendixname}%
2211   \ifdim\jsc@tocl@width<\@tempdima\relax\setlength\jsc@tocl@width{\@tempdima}\fi
2212   \ifdim\jsc@tocl@width<2zw \divide\jsc@tocl@width by 2 \advance\jsc@tocl@width 1zw\fi
2213   \section*{\contentsname}%
2214   \@mkboth{\contentsname}{\contentsname}%
2215 </!book&!report)
2216   \@starttoc{toc}%
2217 (book | report) \if@restonecol\twocolumn\fi
2218 }
```

`\l@part` 部の目次です。

```
2219 \newcommand*{\l@part}[2]{%
2220   \ifnum \c@tocdepth >-2\relax
2221 (!book&!report) \addpenalty\@secpenalty
2222 (book | report) \addpenalty{-\@highpenalty}%
2223   \addvspace{2.25em \@plus\jsc@mpt}%
2224   \begingroup
2225     \parindent \z@
2226 % \@pnumwidth should be \@tocrmarg
2227 % \rightskip \@pnumwidth
2228 \rightskip \@tocrmarg
2229 \parfillskip -\rightskip
2230 {\leavevmode
2231   \large \headfont
2232   \setlength\@lnumwidth{4zw}%
2233   #1\hfil \hb@xt@\@pnumwidth{\hss #2}}\par
2234 \nobreak
```

```

2235 <book | report> \global\@nobreaktrue
2236 <book | report> \everypar{\global\@nobreakfalse\everypar{}}%
2237 \endgroup
2238 \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)
2239 <*book | report>
2240 \newcommand*{\l@chapter}[2]{%
2241 \ifnum \c@tocdepth >\m@ne
2242 \addpenalty{-\@highpenalty}%
2243 \addvspace{1.0em \@plus\jsc@empt}%
2244 % \vskip 1.0em \@plus\p@ % book.cls では↑がこうなっている
2245 \begingroup
2246 \parindent\z@
2247 % \rightskip\@pnumwidth
2248 \rightskip\@tocrmarg
2249 \parfillskip-\rightskip
2250 \leavevmode\headfont
2251 % \if@english\setlength\@lnumwidth{5.5em}\else\setlength\@lnumwidth{4.683zw}\fi
2252 \setlength\@lnumwidth{\jsc@tocl@width}\advance\@lnumwidth 2.683zw
2253 \advance\leftskip\@lnumwidth \hskip-\leftskip
2254 #1\nobreak\hfil\nobreak\hbox to\@pnumwidth{\hss#2}\par
2255 \penalty\@highpenalty
2256 \endgroup
2257 \fi}
2258 </book | report>

```

\l@section 節の目次です。

```

2259 <!*book&!report>
2260 \newcommand*{\l@section}[2]{%
2261 \ifnum \c@tocdepth >\z@
2262 \addpenalty{\@secpenalty}%
2263 \addvspace{1.0em \@plus\jsc@empt}%
2264 \begingroup
2265 \parindent\z@
2266 % \rightskip\@pnumwidth
2267 \rightskip\@tocrmarg
2268 \parfillskip-\rightskip
2269 \leavevmode\headfont
2270 %\setlength\@lnumwidth{4zw}% 元 1.5em [2003-03-02]
2271 \setlength\@lnumwidth{\jsc@tocl@width}\advance\@lnumwidth 2zw
2272 \advance\leftskip\@lnumwidth \hskip-\leftskip
2273 #1\nobreak\hfil\nobreak\hbox to\@pnumwidth{\hss#2}\par
2274 \endgroup
2275 \fi}
2276 </!*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。

```
2277 (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@subsection 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] こども \jsc@tocl@width から決めるようにしてみました。(by ts)
```

```
\l@subparagraph 2278 (*!book&!report)
```

```
2279 % \newcommand*\l@subsection}{\@dottedtocline{2}{1.5em}{2.3em}}
```

```
2280 % \newcommand*\l@subsubsection}{\@dottedtocline{3}{3.8em}{3.2em}}
```

```
2281 % \newcommand*\l@paragraph}{\@dottedtocline{4}{7.0em}{4.1em}}
```

```
2282 % \newcommand*\l@subparagraph}{\@dottedtocline{5}{10em}{5em}}
```

```
2283 %
```

```
2284 % \newcommand*\l@subsection}{\@dottedtocline{2}{1zw}{3zw}}
```

```
2285 % \newcommand*\l@subsubsection}{\@dottedtocline{3}{2zw}{3zw}}
```

```
2286 % \newcommand*\l@paragraph}{\@dottedtocline{4}{3zw}{3zw}}
```

```
2287 % \newcommand*\l@subparagraph}{\@dottedtocline{5}{4zw}{3zw}}
```

```
2288 %
```

```
2289 \newcommand*\l@subsection}{%
```

```
2290     \@tempdima\jsc@tocl@width \advance\@tempdima -1zw
```

```
2291     \@dottedtocline{2}{\@tempdima}{3zw}}
```

```
2292 \newcommand*\l@subsubsection}{%
```

```
2293     \@tempdima\jsc@tocl@width \advance\@tempdima 0zw
```

```
2294     \@dottedtocline{3}{\@tempdima}{4zw}}
```

```
2295 \newcommand*\l@paragraph}{%
```

```
2296     \@tempdima\jsc@tocl@width \advance\@tempdima 1zw
```

```
2297     \@dottedtocline{4}{\@tempdima}{5zw}}
```

```
2298 \newcommand*\l@subparagraph}{%
```

```
2299     \@tempdima\jsc@tocl@width \advance\@tempdima 2zw
```

```
2300     \@dottedtocline{5}{\@tempdima}{6zw}}
```

```
2301 \!/book&!report)
```

```
2302 (*book|report)
```

```
2303 % \newcommand*\l@subsection}{\@dottedtocline{2}{3.8em}{3.2em}}
```

```
2304 % \newcommand*\l@subsubsection}{\@dottedtocline{3}{7.0em}{4.1em}}
```

```
2305 % \newcommand*\l@paragraph}{\@dottedtocline{4}{10em}{5em}}
```

```
2306 % \newcommand*\l@subparagraph}{\@dottedtocline{5}{12em}{6em}}
```

```
2307 \newcommand*\l@section}{%
```

```
2308     \@tempdima\jsc@tocl@width \advance\@tempdima -1zw
```

```
2309     \@dottedtocline{1}{\@tempdima}{3.683zw}}
```

```
2310 \newcommand*\l@subsection}{%
```

```
2311     \@tempdima\jsc@tocl@width \advance\@tempdima 2.683zw
```

```
2312     \@dottedtocline{2}{\@tempdima}{3.5zw}}
```

```
2313 \newcommand*\l@subsubsection}{%
```

```
2314     \@tempdima\jsc@tocl@width \advance\@tempdima 6.183zw
```

```
2315     \@dottedtocline{3}{\@tempdima}{4.5zw}}
```

```
2316 \newcommand*\l@paragraph}{%
```

```
2317     \@tempdima\jsc@tocl@width \advance\@tempdima 10.683zw
```

```
2318     \@dottedtocline{4}{\@tempdima}{5.5zw}}
```

```

2319 \newcommand*{\l@subparagraph}{%
2320     \@tempdima\jsc@tocl@width \advance\@tempdima 16.183zw
2321     \@dottedtocline{5}{\@tempdima}{6.5zw}}
2322 (/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  
`\@lnumwidth` 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。

```

2323 \newdimen\@lnumwidth
2324 \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  
`\jsTocLine` 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。

```

2325 \def\jsTocLine{\leaders\hbox{%
2326     $\m@th \mkern \@dotsep mu\hbox{.}\mkern \@dotsep mu$}\hfill}
2327 \def\@dottedtocline#1#2#3#4#5{\ifnum #1>\c@tocdepth \else
2328     \vskip \z@ \@plus.2\jsc@mpt
2329     {\leftskip #2\relax \rightskip \@tocrmarg \parfillskip -\rightskip
2330     \parindent #2\relax\@afterindenttrue
2331     \interlinepenalty\@M
2332     \leavevmode
2333     \@lnumwidth #3\relax
2334     \advance\leftskip \@lnumwidth \null\nobreak\hskip -\leftskip
2335     {#4}\nobreak
2336     \jsTocLine \nobreak\hb@xt@\@pnumwidth{%
2337     \hfil\normalfont \normalcolor #5}\par}\fi}

```

## ■ 図目次と表目次

`\listoffigures` 図目次を出力します。

```

2338 \newcommand{\listoffigures}{%
2339 (*book | report)
2340     \if@twocolumn\@restonecoltrue\onecolumn
2341     \else\@restonecolfalse\fi
2342     \chapter*{\listfigurename}%
2343     \@mkboth{\listfigurename}{}%
2344 (/book | report)
2345 (*!book&!report)
2346     \section*{\listfigurename}%
2347     \@mkboth{\listfigurename}{\listfigurename}%

```

```

2348 </!book&!report)
2349 \starttoc{lof}%
2350 (book | report) \if@restonecol\twocolumn\fi
2351 }

```

`\l@figure` 図目次の項目を出力します。

```
2352 \newcommand*\l@figure{\@dottedtocline{1}{1zw}{3.683zw}}
```

`\listoftables` 表目次を出力します。

```

2353 \newcommand\listoftables{%
2354 (*book | report)
2355 \if@twocolumn\@restonecoltrue\onecolumn
2356 \else\@restonecolfalse\fi
2357 \chapter*\listtablename}%
2358 \@mkboth{\listtablename}{}%
2359 </book | report)
2360 (*!book&!report)
2361 \section*\listtablename}%
2362 \@mkboth{\listtablename}{\listtablename}%
2363 </!book&!report)
2364 \starttoc{lot}%
2365 (book | report) \if@restonecol\twocolumn\fi
2366 }

```

`\l@table` 表目次は図目次と同じです。

```
2367 \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。

```

2368 \newdimen\bibindent
2369 \setlength\bibindent{2zw}

```

`thebibliography` 参考文献リストを出力します。

```

2370 \newenvironment{thebibliography}[1]{%
2371 \global\let\presectionname\relax
2372 \global\let\postsectionname\relax
2373 (article | jspf) \section*\refname\@mkboth{\refname}{\refname}%
2374 (*kiyou)
2375 \vspace{1.5\baselineskip}
2376 \subsubsection*\refname\@mkboth{\refname}{\refname}%
2377 \vspace{0.5\baselineskip}
2378 </kiyou)
2379 (book | report) \chapter*\bibname\@mkboth{\bibname}{}%
2380 (book | report) \addcontentsline{toc}{chapter}{\bibname}%
2381 \list{\@biblabel{\@arabic\c@enumiv}}%
2382 {\settowidth\labelwidth{\@biblabel{#1}}%
2383 \leftmargin\labelwidth

```

```

2384 \advance\leftmargin\labelsep
2385 \@openbib@code
2386 \usecounter{enumiv}%
2387 \let\p@enumiv\@empty
2388 \renewcommand\theenumiv{\@arabic\c@enumiv}%
2389 (kiyou) \small
2390 \sloppy
2391 \clubpenalty4000
2392 \@clubpenalty\clubpenalty
2393 \widowpenalty4000%
2394 \sfcode'\.\@m}
2395 {\def\@noitemerr
2396 {\@latex@warning{Empty 'thebibliography' environment}}}%
2397 \endlist}

```

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

```
2398 \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。

```
2399 \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。

```
2400 % \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、必要に応じて生かしてください。かっ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~\cite{knu}` のように半角空白で囲んでください。

```

2401 % \def\@citex[#1]#2{\leavevmode
2402 % \let\@citea\@empty
2403 % \@cite{\@for\@citeb:=#2\do
2404 % {\@citea\def\@citea{, \inhibitglue\penalty\@m\ }%
2405 % \edef\@citeb{\expandafter\@firstofone\@citeb\@empty}}%
2406 % \if@files\immediate\write\@auxout{\string\citation{\@citeb}}\fi
2407 % \@ifundefined{b@\@citeb}{\mbox{\normalfont\bfseries ?}}%
2408 % \G@refundefinedtrue
2409 % \@latex@warning
2410 % {Citation '\@citeb' on page \thepage \space undefined}}%
2411 % {\@cite#1\ofmt{\csname b@\@citeb\endcsname}}}{#1}}
2412 % \def\@cite#1#2{\inhibitglue [#1\if@tempwaf, #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。

```

2413 % \DeclareRobustCommand\cite{\unskip
2414 % \@ifnextchar [{\@tempwattrue\@citex}{\@tempwafalse\@citex[]}]

```

```

2415 % \def\@cite#1#2{${\hbox{\scriptsize{#1}\if@tempswa
2416 %   , \inhibitglue\ #2\fi} )}}$}

```

## 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)。

```

2417 \newenvironment{theindex}{% 索引を 3 段組で出力する環境
2418   \if@twocolumn
2419     \onecolumn\@restonecolfalse
2420   \else
2421     \clearpage\@restonecoltrue
2422   \fi
2423   \columnseprule.4pt \columnsep 2zw
2424   \ifx\multicols\undefined
2425 (book | report)   \twocolumn[\@makeschapterhead{\indexname}]%
2426 (book | report)   \addcontentsline{toc}{chapter}{\indexname}]%
2427 (!book&!report)  \def\presectionname{}\def\postsectionname{}%
2428 (!book&!report)  \twocolumn[\section*{\indexname}]%
2429   \else
2430     \ifdim\textwidth<\fullwidth
2431       \setlength{\evensidemargin}{\oddsidemargin}
2432       \setlength{\textwidth}{\fullwidth}
2433       \setlength{\linewidth}{\fullwidth}
2434 (book | report)   \begin{multicols}{3}[\chapter*{\indexname}]%
2435 (book | report)   \addcontentsline{toc}{chapter}{\indexname}]%
2436 (!book&!report)  \def\presectionname{}\def\postsectionname{}%
2437 (!book&!report)  \begin{multicols}{3}[\section*{\indexname}]%
2438   \else
2439 (book | report)   \begin{multicols}{2}[\chapter*{\indexname}]%
2440 (book | report)   \addcontentsline{toc}{chapter}{\indexname}]%
2441 (!book&!report)  \def\presectionname{}\def\postsectionname{}%
2442 (!book&!report)  \begin{multicols}{2}[\section*{\indexname}]%
2443   \fi
2444   \fi
2445 (book | report)   \@mkboth{\indexname}{}%
2446 (!book&!report)  \@mkboth{\indexname}{\indexname}%
2447   \plainifnotempty % \thispagestyle{plain}
2448   \parindent\z@
2449   \parskip\z@ \@plus .3\jcs@mp\relax
2450   \let\item\@idxitem
2451   \raggedright
2452   \footnotesize\narrowbaselines
2453 }{
2454   \ifx\multicols\undefined
2455     \if@restonecol\onecolumn\fi
2456   \else

```

```

2457     \end{multicols}
2458     \fi
2459     \clearpage
2460 }

```

`\@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 2461 \newcommand{\@idxitem}{\par\hangindent 4zw} % 元 40pt
\subsubitem 2462 \newcommand{\subitem}{\@idxitem \hspace*{2zw}} % 元 20pt
2463 \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。

```

2464 \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`, `\alsoname` `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。

```

2465 \newcommand\seename{\if@english see\else → \fi}
2466 \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。

```

2467 \@ifl@t@r\pfmtversion{2016/09/03}
2468     {\jsc@needspace@tchfalse}{\jsc@needspace@tchtrue}
2469 \ifjsc@needspace@tch
2470     \let\footnotes@ve=\footnote
2471     \def\footnote{\inhibitglue\footnotes@ve}
2472     \let\footnotemarks@ve=\footnotemark
2473     \def\footnotemark{\inhibitglue\footnotemarks@ve}
2474 \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。

```

2475 \@ifl@t@r\pfmtversion{2016/04/17}
2476     {\jsc@needspace@tchfalse}{\jsc@needspace@tchtrue}

```

```

2477 \ifjsc@needspace@tch
2478 \renewcommand\@makefnmark{%
2479   \ifydir \hbox{\hbox{\@textsuperscript{\normalfont\@thefnmark}}\hbox{}}%
2480   \else\hbox{\yoko\@textsuperscript{\normalfont\@thefnmark}}\fi}
2481 \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)。

```

2482 \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。

```

2483 % \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。

```

2484 \renewcommand{\footnoterule}{%
2485   \kern-3\jsc@empt
2486   \hrule width .4\columnwidth height 0.4\jsc@empt
2487   \kern 2.6\jsc@empt}

```

`\c@footnote` 脚注番号は章ごとにリセットされます。

```

2488 (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>A</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>A</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>A</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。

```

2489 \long\def\@footnotetext{%
2490   \insert\footins\bgroup
2491     \normalfont\footnotesize
2492     \interlinepenalty\interfootnotelinepenalty
2493     \splittopskip\footnotesep
2494     \splitmaxdepth \dp\strutbox \floatingpenalty \@MM
2495     \hsize\columnwidth \@parboxrestore
2496     \protected@edef\@currentlabel{%
2497       \csname p@footnote\endcsname\@thefnmark
2498     }%

```

```

2499 \color@begingroup
2500 \makefnfntext{%
2501 \rule{z@\footnotesep\ignorespaces}%
2502 \futurelet\jsc@next\jsc@fo@t}
2503 \def\jsc@fo@t{\ifcat\bgroup\noexpand\jsc@next \let\jsc@next\jsc@f@@t
2504 \else \let\jsc@next\jsc@f@t\fi \jsc@next}
2505 \def\jsc@f@@t{\bgroup\aftergroup\jsc@@foot\let\jsc@next}
2506 \def\jsc@f@t#1{#1\jsc@@foot}
2507 \def\jsc@@foot{\@finalstrut\strutbox\color@endgroup\egroup
2508 \ifx\pltx@foot@penalty\undefined\else
2509 \ifhmode\null\fi
2510 \ifnum\pltx@foot@penalty=z@\else
2511 \penalty\pltx@foot@penalty
2512 \pltx@foot@penalty\z@
2513 \fi
2514 \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。

```

2515 \newcommand\makefnfntext[1]{%
2516 \advance\leftskip 3zw
2517 \parindent 1zw
2518 \noindent
2519 \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。

```

2520 % \def\xfootnotenext[#1]{%
2521 % \begingroup
2522 % \ifnum#1>z@
2523 % \csname c@\mpfn\endcsname #1\relax
2524 % \unrestored@protected@xdef\@thefnmark{\thempfn}%
2525 % \else
2526 % \unrestored@protected@xdef\@thefnmark{}%
2527 % \fi
2528 % \endgroup
2529 % \@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` は実はテンポラリーではなく「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)。

```
2530 \def\@inhibitglue{%
2531   \futurelet\@let@token\@inhibitglue}
2532 \begingroup
2533 \let\GDEF=\gdef
2534 \let\CATCODE=\catcode
2535 \let\ENDGROUP=\endgroup
2536 \CATCODE'k=12
2537 \CATCODE'a=12
2538 \CATCODE'n=12
2539 \CATCODE'j=12
2540 \CATCODE'i=12
2541 \CATCODE'c=12
2542 \CATCODE'h=12
2543 \CATCODE'r=12
2544 \CATCODE't=12
2545 \CATCODE'e=12
2546 \GDEF\KANJI@CHARACTER{kanji character }
2547 \ENDGROUP
2548 \def\@inhibitglue{%
2549   \expandafter\expandafter\expandafter\jsc@inhibitglue\expandafter\meaning\expandafter\@let@token
2550 \expandafter\def\expandafter\jsc@inhibitglue\expandafter#\expandafter1\KANJI@CHARACTER#2#3\jsc@
2551   \def\jsc@ig@temp{#1}%
2552   \ifx\jsc@ig@temp@empty
2553     \ifnum\the\inhibitxspcode'#2=2\relax
2554       \inhibitglue
2555     \fi
2556   \fi}
2557 \let\everyparhook=\@inhibitglue
2558 \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。

```
2559 \def\@doendpe{%
2560   \@endpetrue
2561   \def\par{%
2562     \@restorepar\clubpenalty\@clubpenalty\everypar{\everyparhook}\par\@endpefalse}%
2563   \everypar{\setbox\z@\lastbox}\everypar{\everyparhook}\@endpefalse\everyparhook}}
```

[2017-08-31] minipage 環境にも対策します。

```
2564 \def\@setminipage{%
2565   \@minipagetrue
2566   \everypar{\@minipagefalse\everypar{\everyparhook}}%
2567 }
```

\item 命令の直後です。

```
2568 \def\@item[#1]{%
2569   \if@noperitem
2570     \@donoperitem
2571   \else
2572     \if@inlabel
2573       \indent \par
2574     \fi
2575     \ifhmode
2576       \unskip\unskip \par
2577     \fi
2578     \if@newlist
2579       \if@nobreak
2580         \@nbitem
2581       \else
2582         \addpenalty\@beginparpenalty
2583         \addvspace\@topsep
2584         \addvspace{-\parskip}%
2585       \fi
2586     \else
2587       \addpenalty\@itempenalty
2588       \addvspace\itemsep
2589     \fi
2590     \global\@inlabeltrue
2591   \fi
2592   \everypar{%
2593     \@minipagefalse
2594     \global\@newlistfalse
2595     \if@inlabel
2596       \global\@inlabelfalse
2597       {\setbox\z@\lastbox
2598        \ifvoid\z@
2599          \kern-\itemindent
2600        \fi}%
2601     \box\@labels
2602     \penalty\z@
2603   \fi
2604   \if@nobreak
2605     \@nobreakfalse
2606     \clubpenalty \@M
2607   \else
2608     \clubpenalty \@clubpenalty
2609     \everypar{\everyparhook}%
```

```

2610 \fi\everyparhook}%
2611 \if@noitemarg
2612 \@noitemargfalse
2613 \if@nmbrrlist
2614 \refstepcounter\@listctr
2615 \fi
2616 \fi
2617 \sbox\@tempboxa{\makelabel{#1}}%
2618 \global\setbox\@labels\hbox{%
2619 \unhbox\@labels
2620 \hskip \itemindent
2621 \hskip -\labelwidth
2622 \hskip -\labelsep
2623 \ifdim \wd\@tempboxa >\labelwidth
2624 \box\@tempboxa
2625 \else
2626 \hbox to\labelwidth {\unhbox\@tempboxa}%
2627 \fi
2628 \hskip \labelsep}%
2629 \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。

```

2630 \def\@afterheading{%
2631 \@nobraektrue
2632 \everypar{%
2633 \if@nobraek
2634 \@nobraekfalse
2635 \clubpenalty \@M
2636 \if@afterindent \else
2637 {\setbox\z@\lastbox}%
2638 \fi
2639 \else
2640 \clubpenalty \@clubpenalty
2641 \everypar{\everyparhook}%
2642 \fi\everyparhook}}

```

`\@gnewline` についてはちょっと複雑な心境です。もともとの p<sub>TEX</sub> 2<sub>ε</sub> 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。

```

2643 \def\@gnewline #1{%
2644 \ifvmode
2645 \@nolnerr
2646 \else
2647 \unskip \reserved@e {\reserved@f#1}\nobraek \hfil \break \null
2648 \inhibitglue \ignorespaces

```

2649 \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。

\小 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  
\上小 してありますので、コピーします。

```
2650 \if@jslogo
2651   \IfFileExists{jslogo.sty}{%
2652     \RequirePackage{jslogo}%
2653     \def\小{\jslg@small}%
2654     \def\上小{\jslg@uppersmall}%
2655   }{%
2656     \ClassWarningNoLine{\jsc@clsname}{%
2657       The redefinitions of LaTeX-related logos has\MessageBreak
2658       been moved to jslogo.sty since 2016, but\MessageBreak
2659       jslogo.sty not found. Current release of\MessageBreak
2660       'jsclasses' includes it, so please check\MessageBreak
2661       the installation}%
2662   }
2663 \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。

```
2664 %\let\ltx@ifnextchar\@ifnextchar
2665 %\def\ProvidesFile#1{%
2666 %  \begingroup
2667 %    \catcode'\ 10 %
2668 %    \ifnum \endlinechar<256 %
2669 %      \ifnum \endlinechar>\m@ne
2670 %        \catcode\endlinechar 10 %
2671 %    \fi
```

```

2672 % \fi
2673 % \@makeother\/%
2674 % \@makeother\&%
2675 % \ltx@ifnextchar[{\@providesfile{#1}}{\@providesfile{#1}[]}]

```

## 14 初期設定

### ■いろいろな語

```

\prepartname
\postpartname 2676 \newcommand{\prepartname}{\if@english Part~\else 第 \fi}
\prechaptername 2677 \newcommand{\postpartname}{\if@english \else 部 \fi}
2678 (book | report) \newcommand{\prechaptername}{\if@english Chapter~\else 第 \fi}
\postchaptername 2679 (book | report) \newcommand{\postchaptername}{\if@english \else 章 \fi}
\presectionname 2680 \newcommand{\presectionname}{\}% 第
\postsectionname 2681 \newcommand{\postsectionname}{\}% 節

\contentsname
\listfigurename 2682 \newcommand{\contentsname}{\if@english Contents\else 目次 \fi}
\listtablename 2683 \newcommand{\listfigurename}{\if@english List of Figures\else 図目次 \fi}
2684 \newcommand{\listtablename}{\if@english List of Tables\else 表目次 \fi}

\refname
\bibname 2685 \newcommand{\refname}{\if@english References\else 参考文献 \fi}
\indexname 2686 \newcommand{\bibname}{\if@english Bibliography\else 参考文献 \fi}
2687 \newcommand{\indexname}{\if@english Index\else 索引 \fi}

\figurename
\tablename 2688 (!jspf) \newcommand{\figurename}{\if@english Fig.~\else 図 \fi}
2689 (jspf) \newcommand{\figurename}{Fig.~}
2690 (!jspf) \newcommand{\tablename}{\if@english Table~\else 表 \fi}
2691 (jspf) \newcommand{\tablename}{Table~}

\appendixname
\abstractname 2692 % \newcommand{\appendixname}{\if@english Appendix~\else 付録 \fi}
2693 \newcommand{\appendixname}{\if@english \else 付録 \fi}
2694 (!book) \newcommand{\abstractname}{\if@english Abstract\else 概要 \fi}

■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 2021 年 7 月 1 日で、和暦では令和 3 年 7 月 1 日です。

\today
2695 \newif\if 西暦 \西暦 true
2696 \def\西暦{\西暦 true}
2697 \def\和暦{\西暦 false}
2698 \newcount\heisei \heisei\year \advance\heisei-1988\relax

```

```

2699 \def\pltx@today@year@#1{%
2700   \ifnum\numexpr\year-#1=1 元 \else
2701     \ifnum1=\iftdir\ifmdir0\else1\fi\else0\fi
2702       \kansuji\numexpr\year-#1\relax
2703     \else
2704       \number\numexpr\year-#1\relax\nobreak
2705     \fi
2706   \fi 年
2707 }
2708 \def\pltx@today@year{%
2709   \ifnum\numexpr\year*10000+\month*100+\day<19890108
2710     昭和\pltx@today@year@{1925}%
2711   \else\ifnum\numexpr\year*10000+\month*100+\day<20190501
2712     平成\pltx@today@year@{1988}%
2713   \else
2714     令和\pltx@today@year@{2018}%
2715   \fi\fi}
2716 \def\today{%
2717   \if@english
2718     \ifcase\month\or
2719       January\or February\or March\or April\or May\or June\or
2720       July\or August\or September\or October\or November\or December\fi
2721     \space\number\day, \number\year
2722   \else\if 西暦
2723     \ifnum1=\iftdir\ifmdir0\else1\fi\else0\fi \kansuji\year
2724     \else\number\year\nobreak\fi 年
2725   \else
2726     \pltx@today@year
2727   \fi
2728   \ifnum1=\iftdir\ifmdir0\else1\fi\else0\fi
2729     \kansuji\month 月
2730     \kansuji\day 日
2731   \else
2732     \number\month\nobreak 月
2733     \number\day\nobreak 日
2734   \fi\fi}

```

■ハイフネーション例外  $\TeX$  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）

```
2735 \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。

```

2736 <article>\if@slide \pagestyle{empty} \else \pagestyle{plain} \fi
2737 <book>\if@report \pagestyle{plain} \else \pagestyle{headings} \fi
2738 <report | kiyou>\pagestyle{plain}
2739 <jspf>\pagestyle{headings}
2740 \pagenumbering{arabic}
2741 \if@twocolumn

```

```

2742 \twocolumn
2743 \sloppy
2744 \flushbottom
2745 \else
2746 \onecolumn
2747 \raggedbottom
2748 \fi
2749 \if@slide
2750 \renewcommand\kanjifamilydefault{\gtdefault}
2751 \renewcommand\familydefault{\sfdefault}
2752 \raggedright
2753 \xkanjiskip=0.1em\relax
2754 \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 パッケージが読み込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。

```

2755 (*article | book | report)
2756 \@ifpackageloaded{exppl2e}{\jsc@needsp@tchtrue}{\jsc@needsp@tchfalse}
2757 \ifjsc@needsp@tch\else
2758 \expandafter\endinput
2759 \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
2760 \def\@gnewline #1{%
2761   \ifvmode
2762     \@nolnerr
2763   \else
2764     \unskip \reserved@e {\reserved@f#1}\nobreak \hfil \break \hskip \z@
2765     \ignorespaces
2766   \fi}
2767 \</article | book | report)
2768 \</class)

```

以上です。