

X_ƳLaTeX-jā 用 jsclasses 互換ドキュメントクラス

森見幸正

2022/11/06

1 はじめに

これは jsclasses.dtx を X_ƳLaTeX-jā 用に改変したものです。

jsclasses.dtx は LaTeX3 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 TeX 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 TeX 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 upLaTeX 対応パッチを取り込みました。

[X_ƳLaTeX-jā] [2021-09-19] jsclasses 2021/06/28 版ベースで作成しました。

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。

⟨article⟩	xltjsarticle.cls	論文・レポート用
⟨book⟩	xltjsbook.cls	書籍用
⟨report⟩	xltjsreport.cls	レポート用
⟨jspf⟩	xltjspf.cls	某学会誌用
⟨kiyou⟩	xltjskiyou.cls	某紀要用

LaTeX 2_εあるいは pLaTeX 2_ε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 (TeX 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。

[\LaTeX -ja] フォントメトリックは異なります。

■**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、 \TeX 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。

`\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。

```
1 \class
2 \article\def\jsc@clsname{xltjsarticle}
3 \book\def\jsc@clsname{xltjsbook}
4 \report\def\jsc@clsname{xltjsreport}
5 \jpf\def\jsc@clsname{xltjspf}
6 \kiyou\def\jsc@clsname{xltjskiyou}
```

2 オプション

これらのクラスは `\documentclass{xltjsarticle}` あるいは `\documentclass[オプション]{xltjsarticle}` 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。

```
7 \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。

```
8 \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。

```
9 \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。

```
10 \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。

```
11 \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。

[\LaTeX -ja] 数式内で和文は使えません。

```
12 % \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 m^2 、縦横比 $1:\sqrt{2}$ の長方形の辺の長さを mm 単位に切り捨てたものです。これを基準として順に半截しては mm 単位に切り捨てたものが A1, A2, …です。

B 判は JIS と ISO で定義が異なります。JIS では B0 判の面積が 1.5 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。

```

13 \DeclareOption{a3paper}{%
14   \setlength\paperheight {420mm}%
15   \setlength\paperwidth  {297mm}}
16 \DeclareOption{a4paper}{%
17   \setlength\paperheight {297mm}%
18   \setlength\paperwidth  {210mm}}
19 \DeclareOption{a5paper}{%
20   \setlength\paperheight {210mm}%
21   \setlength\paperwidth  {148mm}}
22 \DeclareOption{a6paper}{%
23   \setlength\paperheight {148mm}%
24   \setlength\paperwidth  {105mm}}
25 \DeclareOption{b4paper}{%
26   \setlength\paperheight {364mm}%
27   \setlength\paperwidth  {257mm}}
28 \DeclareOption{b5paper}{%
29   \setlength\paperheight {257mm}%
30   \setlength\paperwidth  {182mm}}
31 \DeclareOption{b6paper}{%
32   \setlength\paperheight {182mm}%
33   \setlength\paperwidth  {128mm}}
34 \DeclareOption{a4j}{%
35   \setlength\paperheight {297mm}%
36   \setlength\paperwidth  {210mm}}
37 \DeclareOption{a5j}{%
38   \setlength\paperheight {210mm}%
39   \setlength\paperwidth  {148mm}}
40 \DeclareOption{b4j}{%
41   \setlength\paperheight {364mm}%
42   \setlength\paperwidth  {257mm}}
43 \DeclareOption{b5j}{%
44   \setlength\paperheight {257mm}%
45   \setlength\paperwidth  {182mm}}
46 \DeclareOption{a4var}{%

```

```

47 \setlength\paperheight {283mm}%
48 \setlength\paperwidth {210mm}}
49 \DeclareOption{b5var}{%
50 \setlength\paperheight {230mm}%
51 \setlength\paperwidth {182mm}}
52 \DeclareOption{letterpaper}{%
53 \setlength\paperheight {11in}%
54 \setlength\paperwidth {8.5in}}
55 \DeclareOption{legalpaper}{%
56 \setlength\paperheight {14in}%
57 \setlength\paperwidth {8.5in}}
58 \DeclareOption{executivepaper}{%
59 \setlength\paperheight {10.5in}%
60 \setlength\paperwidth {7.25in}}

```

■横置き 用紙の縦と横の長さを入れ換えます。

```

61 \newif\if@landscape
62 \@landscapefalse
63 \DeclareOption{landscape}{\@landscapetrue}

```

■slide オプション slide を新設しました。

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

```

64 \newif\if@slide
65 \@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 ヘパッチを当てるオプション nomag* を新設しました。

```

66 \newcommand{\@ptsize}{0}
67 \newif\ifjsc@mag\jsc@magtrue
68 \newif\ifjsc@mag@xreal\jsc@mag@xrealfalse
69 \def\jsc@magscale{1}
70 (*article)
71 \DeclareOption{slide}{%
72 \@slidetrue\def\jsc@magscale{3.583}
73 \renewcommand{\@ptsize}{26}

```

```

74 \@landscapetrue\@titlepagetrue}
75 \</article>
76 \DeclareOption{8pt}{\def\jsc@magscale{0.833}\renewcommand{\@ptsize}{-2}}
77 \DeclareOption{9pt}{\def\jsc@magscale{0.913}\renewcommand{\@ptsize}{-1}}
78 \DeclareOption{10pt}{\def\jsc@magscale{1}\renewcommand{\@ptsize}{0}}
79 \DeclareOption{11pt}{\def\jsc@magscale{1.095}\renewcommand{\@ptsize}{1}}
80 \DeclareOption{12pt}{\def\jsc@magscale{1.200}\renewcommand{\@ptsize}{2}}
81 \DeclareOption{14pt}{\def\jsc@magscale{1.440}\renewcommand{\@ptsize}{4}}
82 \DeclareOption{17pt}{\def\jsc@magscale{1.728}\renewcommand{\@ptsize}{7}}
83 \DeclareOption{20pt}{\def\jsc@magscale{2}\renewcommand{\@ptsize}{10}}
84 \DeclareOption{21pt}{\def\jsc@magscale{2.074}\renewcommand{\@ptsize}{11}}
85 \DeclareOption{25pt}{\def\jsc@magscale{2.488}\renewcommand{\@ptsize}{15}}
86 \DeclareOption{30pt}{\def\jsc@magscale{2.986}\renewcommand{\@ptsize}{20}}
87 \DeclareOption{36pt}{\def\jsc@magscale{3.583}\renewcommand{\@ptsize}{26}}
88 \DeclareOption{43pt}{\def\jsc@magscale{4.300}\renewcommand{\@ptsize}{33}}
89 \DeclareOption{12Q}{\def\jsc@magscale{0.923}\renewcommand{\@ptsize}{1200}}
90 \DeclareOption{14Q}{\def\jsc@magscale{1.077}\renewcommand{\@ptsize}{1400}}
91 \DeclareOption{10ptj}{\def\jsc@magscale{1.085}\renewcommand{\@ptsize}{1001}}
92 \DeclareOption{10.5ptj}{\def\jsc@magscale{1.139}\renewcommand{\@ptsize}{1051}}
93 \DeclareOption{11ptj}{\def\jsc@magscale{1.194}\renewcommand{\@ptsize}{1101}}
94 \DeclareOption{12ptj}{\def\jsc@magscale{1.302}\renewcommand{\@ptsize}{1201}}
95 \DeclareOption{usemag}{\jsc@magtrue\jsc@mag@xrealfalse}
96 \DeclareOption{nomag}{\jsc@magfalse\jsc@mag@xrealfalse}
97 \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 p_{La}T_EX 2_ε 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 p_{La}T_EX 2_ε 本体で宣言されています。

[X_qL_AT_EX-j_a] カウンタ `\hour`, `\minute` や `\iftombow`, `\iftombowdate`, `\@tombowwidth`, `\@bannertoken` はここで宣言します。

```

98 \newcount\hour
99 \newcount\minute
100 \hour\time \divide\hour by 60\relax
101 \@tempcnta\hour \multiply\@tempcnta 60\relax
102 \minute\time \advance\minute-\@tempcnta
103 \newif\iftombow \tombowfalse
104 \newif\iftombowdate \tombowdatetrue
105 \newdimen\@tombowwidth \@tombowwidth.1\p@
106 \newtoks\@bannertoken \@bannertoken{}
107 \DeclareOption{tombow}{%
108   \tombowtrue \tombowdatetrue
109   \setlength{\@tombowwidth}{.1\p@}%
110   \@bannertoken{%
111     \jobname\space(\number\year-\two@digits\month-\two@digits\day
112     \space\two@digits\hour:\two@digits\minute)}}
113 \DeclareOption{tombo}{%

```

```

114 \tombowtrue \tombowdatefalse
115 \setlength{\@tombowwidth}{.1\p@}

```

■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。

```

116 \DeclareOption{mentuke}{%
117 \tombowtrue \tombowdatefalse
118 \setlength{\@tombowwidth}{\z@}

```

■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] `var`twoside でどちらのページも傍注が右側になります。

```

119 \DeclareOption{oneside}{\@twosidefalse \@mparswitchfalse}
120 \DeclareOption{twoside}{\@twosidetrue \@mparswitchtrue}
121 \DeclareOption{var
```

twoside}{\@twosidetrue \@mparswitchfalse}

■二段組 `twocolumn` で二段組になります。

```

122 \DeclareOption{onecolumn}{\@twocolumnfalse}
123 \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。

```

124 \DeclareOption{titlepage}{\@titlepagetrue}
125 \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 \LaTeX の標準クラスが西欧の横組事情しか考慮せずに、奇数ページ起こしと右起こしを一緒にしてしまったせいです。縦組での奇数ページ起こしと横組での偶数ページ起こしも表現したいので、`jsclasses` では新たに `openleft` も追加しました。

```

126 <book | report>\DeclareOption{openright}{\@openrighttrue\@openleftfalse}
127 <book | report>\DeclareOption{openleft}{\@openlefttrue\@openrightfalse}
128 <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` (*env.*) \LaTeX 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。

```

129 \def\eqnarray{%
130 \stepcounter{equation}%
131 \def\@currentlabel{\p@equation\theequation}%
132 \global\@eqnswtrue
133 \m@th
134 \global\@eqcnt\z@

```

```

135 \tabskip\@centering
136 \let\\\@eqncr
137 $$\everycr{}\halign to\displaywidth\bgroup
138 \hskip\@centering$\displaystyle\tabskip\z@skip{##}$\@eqnse1
139 &\global\@eqcnt\@ne \hfil$\displaystyle{ }\##{ }\$ \hfil
140 &\global\@eqcnt\tw@ $\displaystyle{##}$ \hfil\tabskip\@centering
141 &\global\@eqcnt\thr@@ \hb@xt@\z@\bgroup\hss##\egroup
142 \tabskip\z@skip
143 \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。森本さんにしただって訂正しました。

```

144 \DeclareOption{leqno}{\input{leqno.clo}}
145 \DeclareOption{fleqn}{\input{fleqn.clo}}
146 % fleqn 用の eqnarray 環境の再定義
147 \def\eqnarray{%
148 \stepcounter{equation}%
149 \def\@currentlabel{\p@equation\theequation}%
150 \global\@eqnswtrue\m@th
151 \global\@eqcnt\z@
152 \tabskip\mathindent
153 \let\=\@eqncr
154 \setlength\abovedisplayskip{\topsep}%
155 \ifvmode
156 \addtolength\abovedisplayskip{\partopsep}%
157 \fi
158 \addtolength\abovedisplayskip{\parskip}%
159 \setlength\belowdisplayskip{\abovedisplayskip}%
160 \setlength\belowdisplayshortskip{\abovedisplayskip}%
161 \setlength\abovedisplayshortskip{\abovedisplayskip}%
162 $$\everycr{}\halign to\linewidth% $$
163 \bgroup
164 \hskip\@centering$\displaystyle\tabskip\z@skip{##}$\@eqnse1
165 &\global\@eqcnt\@ne \hfil$\displaystyle{ }\##{ }\$ \hfil
166 &\global\@eqcnt\tw@
167 $\displaystyle{##}$ \hfil \tabskip\@centering
168 &\global\@eqcnt\thr@@ \hb@xt@\z@\bgroup\hss##\egroup
169 \tabskip\z@skip\cr
170 }

```

■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。

```

171 % \DeclareOption{openbib}{%
172 % \AtEndOfPackage{%
173 % \renewcommand\@openbib@code{%
174 % \advance\leftmargin\bibindent
175 % \itemindent -\bibindent
176 % \listparindent \itemindent

```

```

177 %      \parsep \z@}%
178 %      \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。AMSTeX 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。

[X_YLaTeX-j_a] 数式内で和文は使えません。

```

179 % \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。

```

180 \DeclareOption{draft}{\setlength\overfullrule{5pt}}
181 \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, dvi_{ps} 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。

[X_YLaTeX-j_a] 標準以外のメトリクスは用意していないのでコメントアウトしておきます。

```

182 \newif\ifmingoth
183 \mingothfalse
184 \newif\ifjisfont
185 \jisfontfalse
186 % \newif\if@jsc@uplatex
187 % \@jsc@uplatexfalse
188 % \newif\if@jsc@autodetect
189 % \@jsc@autodetectfalse
190 % \DeclareOption{winjis}{%
191 %     \ClassWarningNoLine{\jsc@clsname}{%
192 %         The option `winjis' has been removed;\MessageBreak
193 %         Use ``\string\usepackage{winjis}' instead}}
194 % \DeclareOption{mingoth}{\mingothtrue}
195 % \DeclareOption{jis}{\jisfonttrue}
196 % \DeclareOption{uplatex}{\@jsc@uplatextrue}
197 % \DeclareOption{autodetect-engine}{\@jsc@autodetecttrue}

```



```

198 % \def\jsc@JYn{\if@jsc@uplatex JY2\else JY1\fi}
199 % \def\jsc@JTn{\if@jsc@uplatex JT2\else JT1\fi}
200 % \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。

[X_YLaTeX-ja] 常に \pdfpagewidth, \pdfpageheight を設定します。

```

201 % \newif\ifpapersize
202 % \papersizefalse
203 % \DeclareOption{papersize}{\papersizetrue}

```

■英語化 オプション english を新設しました。

```

204 \newif\if@english
205 \@englishfalse
206 \DeclareOption{english}{\@englishttrue}

```

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

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

```

207 <*book>
208 \newif\if@report
209 \@reportfalse
210 \DeclareOption{report}{\@reporttrue\@openrightfalse\@twosidefalse\@mparswitchfalse}
211 </book>

```

■jslogo パッケージの読み込み LaTeX 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。

```

212 \newif\if@jslogo \@jslogotrue
213 \DeclareOption{jslogo}{\@jslogotrue}
214 \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。

```

215 <article>\ExecuteOptions{a4paper,oneside,onecolumn,notitlepage,final}
216 <book>\ExecuteOptions{a4paper,twoside,onecolumn,titlepage,openright,final}
217 <report>\ExecuteOptions{a4paper,oneside,onecolumn,titlepage,openany,final}
218 <jspf>\ExecuteOptions{a4paper,twoside,twocolumn,notitlepage,fleqn,final}
219 <kiyou>\ExecuteOptions{a4paper,twoside,twocolumn,notitlepage,final}
220 \ProcessOptions

後処理
221 \if@slide
222 \def\maybeblue{\ifundefined{ver@color.sty}}{\color{blue}}
223 \fi
224 \if@landscape

```

```

225 \setlength\@tempdima {\paperheight}
226 \setlength\paperheight{\paperwidth}
227 \setlength\paperwidth {\@tempdima}
228 \fi

[XLaTEX-ja] パッケージを読み込みます。
229 \langle!jspf\rangle\def\Cjascale{0.924690}
230 \langlejspf\rangle\def\Cjascale{0.903375}
231 \RequirePackage[jascale=\Cjascale]{xelatexja}

[XLaTEX-ja] パッケージを読み込みます。
232 \iftombow
233 \xltjTombowSetup{%
234     tombow=true,%
235     banner={\iftombowdate\the\@bannertoken\fi},%
236     thickness=\@tombowwidth}
237 \fi

```

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

トンボの付いたときの用紙サイズは無意味ですが、いわゆる「ノビ」サイズという縦横 1 インチずつ長い用紙に出力することを考えて、1 インチずつ加えました。ところが p_{La}T_EX 2_ε 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_AT_EX 2_ε 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)。

[X_{La}T_EX-j_a] papersize special ではなく \pdfpagewidth、\pdfpageheight を使いますが、「\mag にあわせてスケール」よりもあとで設定します。

[X_{La}T_EX-j_a 2022-11-06] L_AT_EX 2_ε 2022-11-01 でカーネルに \stockwidth、\stockheight が追加されたので、クラスファイル側では未定義のときのみこれらの長さ変数を定義します。

```

238 \iftombow
239 \ifx\stockwidth\undefined\newdimen\stockwidth\fi
240 \ifx\stockheight\undefined\newdimen\stockheight\fi

```

```

241 \setlength{\stockwidth}{\paperwidth}
242 \setlength{\stockheight}{\paperheight}
243 \advance \stockwidth 2in
244 \advance \stockheight 2in
245 \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。

```

246 <article | book | report>\if@slide\def\n@baseline{13}\else\def\n@baseline{16}\fi
247 <jspf>\def\n@baseline{14.554375}
248 <kiyou>\def\n@baseline{14.897}

```

■拡大率の設定 サイズの変更は $\text{T}_{\text{E}}\text{X}$ 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@empt` 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。

```

249 \newdimen\jsc@empt
250 \newdimen\jsc@mmm
251 \def\inv@mag{1}
252 \ifjsc@mag
253 \jsc@empt=1\p@
254 \jsc@mmm=1mm
255 \ifnum\@ptsize=-2
256 \mag 833
257 \def\inv@mag{1.20048}
258 \def\n@baseline{15}%
259 \fi
260 \ifnum\@ptsize=-1
261 \mag 913 % formerly 900
262 \def\inv@mag{1.09529}
263 \def\n@baseline{15}%
264 \fi
265 \ifnum\@ptsize=1
266 \mag 1095 % formerly 1100
267 \def\inv@mag{0.913242}
268 \fi
269 \ifnum\@ptsize=2
270 \mag 1200
271 \def\inv@mag{0.833333}

```

```

272 \fi
273 \ifnum\@ptsize=4
274 \mag 1440
275 \def\inv@mag{0.694444}
276 \fi
277 \ifnum\@ptsize=7
278 \mag 1728
279 \def\inv@mag{0.578704}
280 \fi
281 \ifnum\@ptsize=10
282 \mag 2000
283 \def\inv@mag{0.5}
284 \fi
285 \ifnum\@ptsize=11
286 \mag 2074
287 \def\inv@mag{0.48216}
288 \fi
289 \ifnum\@ptsize=15
290 \mag 2488
291 \def\inv@mag{0.401929}
292 \fi
293 \ifnum\@ptsize=20
294 \mag 2986
295 \def\inv@mag{0.334896}
296 \fi
297 \ifnum\@ptsize=26
298 \mag 3583
299 \def\inv@mag{0.279096}
300 \fi
301 \ifnum\@ptsize=33
302 \mag 4300
303 \def\inv@mag{0.232558}
304 \fi
305 \ifnum\@ptsize=1200
306 \mag 923
307 \def\inv@mag{1.0834236}
308 \fi
309 \ifnum\@ptsize=1400
310 \mag 1077
311 \def\inv@mag{0.928505}
312 \fi
313 \ifnum\@ptsize=1001
314 \mag 1085
315 \def\inv@mag{0.921659}
316 \fi
317 \ifnum\@ptsize=1051
318 \mag 1139
319 \def\inv@mag{0.877963}
320 \fi

```

```

321 \ifnum\@ptsize=1101
322 \mag 1194
323 \def\inv@mag{0.837521}
324 \fi
325 \ifnum\@ptsize=1201
326 \mag 1302
327 \def\inv@mag{0.768049}
328 \fi
329 \else
330 \jsc@mpt=\jsc@magscale\p@
331 \jsc@mmm=\jsc@magscale mm
332 \def\inv@mag{1}
333 \ifnum\@ptsize=-2
334 \def\n@baseline{15}%
335 \fi
336 \ifnum\@ptsize=-1
337 \def\n@baseline{15}%
338 \fi
339 \fi
340 <*kiyou>
341 \def\jsc@magscale{0.9769230}
342 \ifjsc@mag
343 \mag 977
344 \def\inv@mag{1.02354}
345 \jsc@mpt=1\p@
346 \jsc@mmm=1mm
347 \else
348 \jsc@mpt=\jsc@magscale\p@
349 \jsc@mmm=\jsc@magscale mm
350 \def\inv@mag{1}
351 \fi
352 </kiyou>
353 \ifjsc@mag@xreal
354 \RequirePackage{type1cm}
355 \mathchardef\jsc@csta=259
356 \def\jsc@invscale#1#2{%
357 \begingroup \@tempdima=#1\relax \@tempdimb#2\p@\relax
358 \@tempcnta\@tempdima \multiply\@tempcnta\@cclvi
359 \divide\@tempcnta\@tempdimb \multiply\@tempcnta\@cclvi
360 \@tempcntb\p@ \divide\@tempcntb\@tempdimb
361 \advance\@tempcnta-\@tempcntb \advance\@tempcnta-\tw@
362 \@tempdimb\@tempcnta\@ne
363 \advance\@tempcnta\@tempcntb \advance\@tempcnta\@tempcntb
364 \advance\@tempcnta\jsc@csta \@tempdimc\@tempcnta\@ne
365 \@whiledim\@tempdimb<\@tempdimc\do{%
366 \@tempcntb\@tempdimb \advance\@tempcntb\@tempdimc
367 \advance\@tempcntb\@ne \divide\@tempcntb\@tw@
368 \ifdim #2\@tempcntb>\@tempdima
369 \advance\@tempcntb\@ne \@tempdimc=\@tempcntb\@ne

```

```

370     \else \@tempdimb=\@tempcntb\@ne \fi}%
371     \xdef\jsc@gtmpa{\the\@tempdimb}%
372     \endgroup #1=\jsc@gtmpa\relax}
373 \expandafter\let\csname OT1/cmr/m/n/10\endcsname\relax
374 \expandafter\let\csname OMX/cmex/m/n/10\endcsname\relax
375 \let\jsc@get@external@font\get@external@font
376 \def\get@external@font{%
377     \jsc@preadjust@extract@font
378     \jsc@get@external@font}
379 \def\jsc@fstrunc#1{%
380     \edef\jsc@tmpa{\strip@pt#1}%
381     \expandafter\jsc@fstrunc@a\jsc@tmpa.****\@nil}
382 \def\jsc@fstrunc@a#1.#2#3#4#5#6\@nil{%
383     \if#5*\else
384         \edef\jsc@tmpa{#1%
385             \ifnum#2#3>\z@ .#2\ifnum#3>\z@ #3\fi\fi}%
386         \fi}
387 \def\jsc@preadjust@extract@font{%
388     \let\jsc@req@size\f@size
389     \dimen@f@size\p@ \jsc@invscale\dimen@\jsc@magscale
390     \advance\dimen@.005pt\relax \jsc@fstrunc\dimen@
391     \let\jsc@ref@size\jsc@tmpa
392     \let\f@size\jsc@ref@size}
393 \def\execute@size@function#1{%
394     \let\jsc@cref@size\f@size
395     \let\f@size\jsc@req@size
396     \csname s@fct@#1\endcsname}
397 \let\jsc@DeclareErrorFont\DeclareErrorFont
398 \def\DeclareErrorFont#1#2#3#4#5{%
399     \@tempdimc#5\p@ \@tempdimc\jsc@magscale\@tempdimc
400     \edef\jsc@tmpa{{#1}{#2}{#3}{#4}{\strip@pt\@tempdimc}}
401     \expandafter\jsc@DeclareErrorFont\jsc@tmpa}
402 \def\gen@sfcnt{%
403     \edef\mandatory@arg{\mandatory@arg\jsc@cref@size}%
404     \empty@sfcnt}
405 \def\genb@sfcnt{%
406     \edef\mandatory@arg{%
407         \mandatory@arg\expandafter\genb@x\jsc@cref@size.. \@@}%
408     \empty@sfcnt}
409 \DeclareErrorFont{OT1}{cmr}{m}{n}{10}
410 \fi

```

[2016-11-16] latex.ltx (ltspace.dtx) で定義されている `\smallskip` の、単位 `pt` を `\jsc@empt` 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` 411 `\def\jsc@smallskip{\vspace\jsc@smallskipamount}`

`\jsc@bigskip` 412 `%\def\jsc@medskip{\vspace\jsc@medskipamount}`

```
413 %\def\jsc@bigskip{\vspace\jsc@bigskipamount}
```

```
\jsc@smallskipamount
```

```
\jsc@medskipamount 414 \newskip\jsc@smallskipamount
```

```
\jsc@bigskipamount 415 \jsc@smallskipamount=3\jsc@mpt plus 1\jsc@mpt minus 1\jsc@mpt
```

```
416 %\newskip\jsc@medskipamount
```

```
417 %\jsc@medskipamount =6\jsc@mpt plus 2\jsc@mpt minus 2\jsc@mpt
```

```
418 %\newskip\jsc@bigskipamount
```

```
419 %\jsc@bigskipamoun =12\jsc@mpt plus 4\jsc@mpt minus 4\jsc@mpt
```

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

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

[2017-01-11] トンボオ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。

```
420 \setlength\paperwidth{\inv@mag\paperwidth}%
```

```
421 \setlength\paperheight{\inv@mag\paperheight}%
```

```
422 \iftombow
```

```
423 \setlength\stockwidth{\inv@mag\stockwidth}%
```

```
424 \setlength\stockheight{\inv@mag\stockheight}%
```

```
425 \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。

```
426 % \ifpapersize
```

```
427 % \setlength{\@tempdima}{\paperwidth}
```

```
428 % \setlength{\@tempdimb}{\paperheight}
```

```
429 % \iftombow
```

```
430 % \advance \@tempdima 2truein
```

```
431 % \advance \@tempdimb 2truein
```

```
432 % \fi
```

```
433 % \AtBeginDvi{\special{pdf: pagesize width \the\@tempdima space height \the\@tempdimb}}
```

```
434 % \fi
```

[X_YL_AT_EX-j_a] \pdfpagewidth, \pdfpageheight を設定します。

```
435 \iftombow
```

```
436 \setlength{\pdfpagewidth}{\stockwidth}
```

```
437 \setlength{\pdfpageheight}{\stockheight}
```

```
438 \else
```

```
439 \setlength{\pdfpagewidth}{\paperwidth}
```

```
440 \setlength{\pdfpageheight}{\paperheight}
```

```
441 \fi
```

3 和文フォントの変更

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

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

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

[2018-02-04] 上記のと通りの「クラスファイルが意図する和文スケール値 (1zw ÷ 要求サイズ)」を表す実数値マクロ `\Cjascal` 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。

[$\mathrm{X}_{\mathrm{L}}\mathrm{A}_{\mathrm{T}}\mathrm{E}_{\mathrm{X}}\text{-ja}$] `xelatexja.sty` 読み込み時に指定しています。

```
442 <!*jspf>
443 % \ifmingoth
444 % \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.961] \jsc@pfx@ min10}{ }
445 % \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.961] \jsc@pfx@ goth10}{ }
446 % \DeclareFontShape{\jsc@JTn}{mc}{m}{n}{<-> s * [0.961] \jsc@pfx@ tmin10}{ }
447 % \DeclareFontShape{\jsc@JTn}{gt}{m}{n}{<-> s * [0.961] \jsc@pfx@ tgoth10}{ }
448 % \else
449 % \ifjisfont
450 % \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.961] \jsc@pfx@ jis}{ }
451 % \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.961] \jsc@pfx@ jisg}{ }
452 % \DeclareFontShape{\jsc@JTn}{mc}{m}{n}{<-> s * [0.961] \jsc@pfx@ tmin10}{ }
453 % \DeclareFontShape{\jsc@JTn}{gt}{m}{n}{<-> s * [0.961] \jsc@pfx@ tgoth10}{ }
454 % \else
455 % \if@jsc@uplatex
456 % \DeclareFontShape{JY2}{mc}{m}{n}{<-> s * [0.924690] upjisr-h}{ }
457 % \DeclareFontShape{JY2}{gt}{m}{n}{<-> s * [0.924690] upjisg-h}{ }
458 % \DeclareFontShape{JT2}{mc}{m}{n}{<-> s * [0.924690] upjisr-v}{ }
459 % \DeclareFontShape{JT2}{gt}{m}{n}{<-> s * [0.924690] upjisg-v}{ }
460 % \else
461 % \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.961] \jsc@pfx@ jis}{ }
462 % \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.961] \jsc@pfx@ jisg}{ }
463 % \DeclareFontShape{\jsc@JTn}{mc}{m}{n}{<-> s * [0.961] \jsc@pfx@ tmin10}{ }
```



```

464 % \DeclareFontShape{\jsc@JTn}{gt}{m}{n}{<-> s * [0.961] \jsc@pfx@ tgoth10}{ }
465 % \fi
466 % \fi
467 % \fi
468 </!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。

[X_YLaTeX-ja] xelatexja.sty 読み込み時に指定しています。

```

469 <*jspf>
470 % \ifmingoth
471 % \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ min10}{ }
472 % \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ goth10}{ }
473 % \DeclareFontShape{\jsc@JTn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tmin10}{ }
474 % \DeclareFontShape{\jsc@JTn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tgoth10}{ }
475 % \else
476 % \ifjisfont
477 % \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ jis}{ }
478 % \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ jisg}{ }
479 % \DeclareFontShape{\jsc@JTn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tmin10}{ }
480 % \DeclareFontShape{\jsc@JTn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tgoth10}{ }
481 % \else
482 % \if@jsc@uplatex
483 % \DeclareFontShape{JY2}{mc}{m}{n}{<-> s * [0.903375] upjisr-h}{ }
484 % \DeclareFontShape{JY2}{gt}{m}{n}{<-> s * [0.903375] upjisg-h}{ }
485 % \DeclareFontShape{JT2}{mc}{m}{n}{<-> s * [0.903375] upjisr-v}{ }
486 % \DeclareFontShape{JT2}{gt}{m}{n}{<-> s * [0.903375] upjisg-v}{ }
487 % \else
488 % \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ jis}{ }
489 % \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ jisg}{ }
490 % \DeclareFontShape{\jsc@JTn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tmin10}{ }
491 % \DeclareFontShape{\jsc@JTn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tgoth10}{ }
492 % \fi
493 % \fi
494 % \fi
495 </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、T_EX 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。

[X_YLaTeX-j_a] NFSS 周りではできてません。

```
496 % \DeclareFontShape{\jsc@JYn}{mc}{bx}{n}{<->ssub*gt/m/n}{ } % in \jsc@JYnmc
497 % \DeclareFontShape{\jsc@JYn}{gt}{bx}{n}{<->ssub*gt/m/n}{ } % in \jsc@JYngt
498 % \DeclareFontShape{\jsc@JYn}{mc}{m}{it}{<->ssub*mc/m/n}{ }
499 % \DeclareFontShape{\jsc@JYn}{mc}{m}{sl}{<->ssub*mc/m/n}{ }
500 % \DeclareFontShape{\jsc@JYn}{mc}{m}{sc}{<->ssub*mc/m/n}{ }
501 % \DeclareFontShape{\jsc@JYn}{gt}{m}{it}{<->ssub*gt/m/n}{ }
502 % \DeclareFontShape{\jsc@JYn}{gt}{m}{sl}{<->ssub*gt/m/n}{ }
503 % \DeclareFontShape{\jsc@JYn}{mc}{bx}{it}{<->ssub*gt/m/n}{ }
504 % \DeclareFontShape{\jsc@JYn}{mc}{bx}{sl}{<->ssub*gt/m/n}{ }
505 % \DeclareFontShape{\jsc@JTn}{mc}{bx}{n}{<->ssub*gt/m/n}{ } % in \jsc@JTnmc
506 % \DeclareFontShape{\jsc@JTn}{gt}{bx}{n}{<->ssub*gt/m/n}{ } % in \jsc@JTngt
507 % \DeclareFontShape{\jsc@JTn}{mc}{m}{it}{<->ssub*mc/m/n}{ }
508 % \DeclareFontShape{\jsc@JTn}{mc}{m}{sl}{<->ssub*mc/m/n}{ }
509 % \DeclareFontShape{\jsc@JTn}{mc}{m}{sc}{<->ssub*mc/m/n}{ }
510 % \DeclareFontShape{\jsc@JTn}{gt}{m}{it}{<->ssub*gt/m/n}{ }
511 % \DeclareFontShape{\jsc@JTn}{gt}{m}{sl}{<->ssub*gt/m/n}{ }
512 % \DeclareFontShape{\jsc@JTn}{mc}{bx}{it}{<->ssub*gt/m/n}{ }
513 % \DeclareFontShape{\jsc@JTn}{mc}{bx}{sl}{<->ssub*gt/m/n}{ }
```

[2020-02-02] LaTeX 2_ε 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 LaTeX 2_ε 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] LaTeX 2_ε 2020-10-01 では `\AddToHook` を利用します。

```
514 \AddToHook{rmfamily}{\mcfamily}
515 \AddToHook{sffamily}{\gtfamily}
516 \AddToHook{ttfamily}{\gtfamily}
```

`\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が、変わらなくなっていましたので、修正しました。

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

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

```
517 \ifx\DeclareFixJFMCJKTextFontCommand\@undefined
```

```

518 \DeclareRobustCommand\textmc[1]{%
519   \relax\ifmmode \expandafter\nfss@text \fi{\mcfamily #1}}
520 \DeclareRobustCommand\textgt[1]{%
521   \relax\ifmmode \expandafter\nfss@text \fi{\gtfamily #1}}
522 \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。

[X_YLaTeX-j_a] 数式内で和文は使えません。

```

523 % \def\reDeclareMathAlphabet#1#2#3{%
524   \edef\@tempa{\expandafter\@gobble\string#2}%
525   \edef\@tempb{\expandafter\@gobble\string#3}%
526   \edef\@tempc{\string @\expandafter\@gobbletwo\string#2}%
527   \ifx\@tempc\@tempa%
528     \edef\@tempa{\expandafter\@gobbletwo\string#2}%
529     \edef\@tempb{\expandafter\@gobbletwo\string#3}%
530   \fi
531   \begingroup
532     \let\protect\noexpand
533     \def\@tempaa{\relax}%
534     \expandafter\ifx\csname RDMAorg@\@tempa\endcsname\relax
535       \edef\@tempaa{\expandafter\def\expandafter\noexpand%
536         \csname RDMAorg@\@tempa\endcsname{%
537           \expandafter\noexpand\csname\@tempa\endcsname}}%
538     \fi
539     \def\@tempbb{\relax}%
540     \expandafter\ifx\csname RDMAorg@\@tempb\endcsname\relax
541       \edef\@tempbb{\expandafter\def\expandafter\noexpand%
542         \csname RDMAorg@\@tempb\endcsname{%
543           \expandafter\noexpand\csname\@tempb\endcsname}}%
544     \fi
545     \edef\@tempc{\@tempaa\@tempbb}%
546     \expandafter\endgroup\@tempc%
547     \edef#1{\noexpand\protect\expandafter\noexpand\csname%
548       \expandafter\@gobble\string#1\space\space\endcsname}%
549     \expandafter\edef\csname\expandafter\@gobble\string#1\space\space\endcsname%
550       {\noexpand\DualLang@mathalph@bet%
551         {\expandafter\noexpand\csname RDMAorg@\@tempa\endcsname}%
552         {\expandafter\noexpand\csname RDMAorg@\@tempb\endcsname}%
553       }%
554   \}
555 % \@onlypreamble\reDeclareMathAlphabet
556 % \def\DualLang@mathalph@bet#1#2{%

```

```

557 % \relax\ifmmode
558 % \ifx\math@bgroup\bgroup% 2e normal style (\mathrm{...})
559 % \bgroup\let\DualLang@Mfontsw\DLMfontsw@standard
560 % \else
561 % \ifx\math@bgroup\relax% 2e two letter style (\rm->\mathrm)
562 % \let\DualLang@Mfontsw\DLMfontsw@oldstyle
563 % \else
564 % \ifx\math@bgroup@empty% 2.09 oldlfont style ({\mathrm ...})
565 % \let\DualLang@Mfontsw\DLMfontsw@oldlfont
566 % \else% panic! assume 2e normal style
567 % \bgroup\let\DualLang@Mfontsw\DLMfontsw@standard
568 % \fi
569 % \fi
570 % \fi
571 % \else
572 % \let\DualLang@Mfontsw@firstoftwo
573 % \fi
574 % \DualLang@Mfontsw{#1}{#2}%
575 % }
576 % \def\DLMfontsw@standard#1#2#3{#1{#2{#3}}\egroup}
577 % \def\DLMfontsw@oldstyle#1#2{#1\relax\@fontswitch\relax{#2}}
578 % \def\DLMfontsw@oldlfont#1#2{#1\relax#2\relax}
579 % \if@enablejfam
580 % \DeclareSymbolFont{mincho}{\jsc@JYn}{mc}{m}{n}
581 % \DeclareSymbolFontAlphabet{\mathmc}{mincho}
582 % \SetSymbolFont{mincho}{bold}{\jsc@JYn}{gt}{m}{n}
583 % \jfam\symmincho
584 % \DeclareMathAlphabet{\mathgt}{\jsc@JYn}{gt}{m}{n}
585 % \AtBeginDocument{%
586 % \reDeclareMathAlphabet{\mathrm}{\@mathrm}{\@mathmc}
587 % \reDeclareMathAlphabet{\mathbf}{\@mathbf}{\@mathgt}}
588 % \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。

```
589 % \DeclareTextCommand{\textsterling}{OT1}{\itshape\char`\$}}
```

禁則パラメータも若干修正します。

[X_YLaTeX-ja] 禁則パラメータは文字単位でなくクラス単位でしか指定できないのでコメントアウトしておきます。

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。

```

590 % \prebreakpenalty\jis"2147=10000      % 5000  '
591 % \postbreakpenalty\jis"2148=10000      % 5000  "
592 % \prebreakpenalty\jis"2149=10000      % 5000  "

```

「 $\mathrm{T}_\mathrm{E}\mathrm{X}$!」「 π 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。

```

593 % \inhibitxspcode`!=1
594 % \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

```

595 % \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. $\mathrm{mbox}\{\}$ 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。

```

596 % \xspcode`+=3
597 % \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。

```

598 % \xspcode`^^80=3
599 % \xspcode`^^81=3
600 % \xspcode`^^82=3
601 % \xspcode`^^83=3
602 % \xspcode`^^84=3
603 % \xspcode`^^85=3
604 % \xspcode`^^86=3
605 % \xspcode`^^87=3
606 % \xspcode`^^88=3
607 % \xspcode`^^89=3
608 % \xspcode`^^8a=3
609 % \xspcode`^^8b=3
610 % \xspcode`^^8c=3
611 % \xspcode`^^8d=3
612 % \xspcode`^^8e=3
613 % \xspcode`^^8f=3
614 % \xspcode`^^90=3
615 % \xspcode`^^91=3
616 % \xspcode`^^92=3
617 % \xspcode`^^93=3
618 % \xspcode`^^94=3
619 % \xspcode`^^95=3
620 % \xspcode`^^96=3
621 % \xspcode`^^97=3
622 % \xspcode`^^98=3
623 % \xspcode`^^99=3
624 % \xspcode`^^9a=3

```

625 % \xspcode`^^9b=3
626 % \xspcode`^^9c=3
627 % \xspcode`^^9d=3
628 % \xspcode`^^9e=3
629 % \xspcode`^^9f=3
630 % \xspcode`^^a0=3
631 % \xspcode`^^a1=3
632 % \xspcode`^^a2=3
633 % \xspcode`^^a3=3
634 % \xspcode`^^a4=3
635 % \xspcode`^^a5=3
636 % \xspcode`^^a6=3
637 % \xspcode`^^a7=3
638 % \xspcode`^^a8=3
639 % \xspcode`^^a9=3
640 % \xspcode`^^aa=3
641 % \xspcode`^^ab=3
642 % \xspcode`^^ac=3
643 % \xspcode`^^ad=3
644 % \xspcode`^^ae=3
645 % \xspcode`^^af=3
646 % \xspcode`^^b0=3
647 % \xspcode`^^b1=3
648 % \xspcode`^^b2=3
649 % \xspcode`^^b3=3
650 % \xspcode`^^b4=3
651 % \xspcode`^^b5=3
652 % \xspcode`^^b6=3
653 % \xspcode`^^b7=3
654 % \xspcode`^^b8=3
655 % \xspcode`^^b9=3
656 % \xspcode`^^ba=3
657 % \xspcode`^^bb=3
658 % \xspcode`^^bc=3
659 % \xspcode`^^bd=3
660 % \xspcode`^^be=3
661 % \xspcode`^^bf=3
662 % \xspcode`^^c0=3
663 % \xspcode`^^c1=3
664 % \xspcode`^^c2=3
665 % \xspcode`^^c3=3
666 % \xspcode`^^c4=3
667 % \xspcode`^^c5=3
668 % \xspcode`^^c6=3
669 % \xspcode`^^c7=3
670 % \xspcode`^^c8=3
671 % \xspcode`^^c9=3
672 % \xspcode`^^ca=3
673 % \xspcode`^^cb=3

674 % \xspcode`^^cc=3
 675 % \xspcode`^^cd=3
 676 % \xspcode`^^ce=3
 677 % \xspcode`^^cf=3
 678 % \xspcode`^^d0=3
 679 % \xspcode`^^d1=3
 680 % \xspcode`^^d2=3
 681 % \xspcode`^^d3=3
 682 % \xspcode`^^d4=3
 683 % \xspcode`^^d5=3
 684 % \xspcode`^^d6=3
 685 % \xspcode`^^d7=3
 686 % \xspcode`^^d8=3
 687 % \xspcode`^^d9=3
 688 % \xspcode`^^da=3
 689 % \xspcode`^^db=3
 690 % \xspcode`^^dc=3
 691 % \xspcode`^^dd=3
 692 % \xspcode`^^de=3
 693 % \xspcode`^^df=3
 694 % \xspcode`^^e0=3
 695 % \xspcode`^^e1=3
 696 % \xspcode`^^e2=3
 697 % \xspcode`^^e3=3
 698 % \xspcode`^^e4=3
 699 % \xspcode`^^e5=3
 700 % \xspcode`^^e6=3
 701 % \xspcode`^^e7=3
 702 % \xspcode`^^e8=3
 703 % \xspcode`^^e9=3
 704 % \xspcode`^^ea=3
 705 % \xspcode`^^eb=3
 706 % \xspcode`^^ec=3
 707 % \xspcode`^^ed=3
 708 % \xspcode`^^ee=3
 709 % \xspcode`^^ef=3
 710 % \xspcode`^^f0=3
 711 % \xspcode`^^f1=3
 712 % \xspcode`^^f2=3
 713 % \xspcode`^^f3=3
 714 % \xspcode`^^f4=3
 715 % \xspcode`^^f5=3
 716 % \xspcode`^^f6=3
 717 % \xspcode`^^f7=3
 718 % \xspcode`^^f8=3
 719 % \xspcode`^^f9=3
 720 % \xspcode`^^fa=3
 721 % \xspcode`^^fb=3
 722 % \xspcode`^^fc=3

```

723 % \xspcode^^fd=3
724 % \xspcode^^fe=3
725 % \xspcode^^ff=3

```

\@ 欧文といえば， \LaTeX の $\text{\def\@{\spacefactor\@m}}$ という定義（ \@m は 1000）では 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， I watch TV.\@ と書くことにします。

[2016-07-14] 2015-01-01 の \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。

```

726 \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 \LaTeX 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， \LaTeX 本体で定義されています。

\@vpt	5	\@vipt	6	\@viipt	7
\@viiipt	8	\@ixpt	9	\@xpt	10
\@xipt	10.95	\@xiipt	12	\@xivpt	14.4

\@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 $\text{p\LaTeX 2}_\epsilon$ で $\text{0pt plus .4pt minus .5pt}$ 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。

```

727 %% \@setfontsize with \parindent and \(\x)kanjiskip settings

```



```

728 \def\@setfontsize#1#2#3{%
729   \ifx\protect\@typeset@protect
730     \let\@currsz#1%
731   \fi
732   \fontsize{#2}{#3}\selectfont
733   \ifdim\parindent>\z@
734     \if@english
735       \parindent=1em
736     \else
737       \parindent=1\zw
738     \fi
739   \fi
740   \setkanjiskip{0\zw plus .1\zw minus .01\zw}%
741   \@tempskipa\getxkanjiskip\relax
742   \ifdim\@tempskipa>\z@
743     \if@slide
744       \setxkanjiskip{0.1em}%
745     \else
746       \setxkanjiskip{0.25em plus 0.15em minus 0.06em}%
747     \fi
748   \fi
749 }

```

`\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。

```

750 \def\jsc@setfontsize#1#2#3{%
751   \@setfontsize#1{#2\jsc@mpt}{#3\jsc@mpt}}

```

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

```

752 \emergencystretch 3\zw

```

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

`\narrowbaselines` [2003-06-30] 数式に入るところで `\narrowbaselines` を実行しているので `\abovedisplayskip` `\widebaselines` 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。

```

753 \newif\ifnarrowbaselines
754 \if@english
755   \narrowbaselinestrue
756 \fi
757 \def\narrowbaselines{%
758   \narrowbaselinestrue
759   \skip0=\abovedisplayskip
760   \skip2=\abovedisplayshortskip
761   \skip4=\belowdisplayskip

```

```

762 \skip6=\belowdisplayshortskip
763 \@currsize\selectfont
764 \abovedisplayskip=\skip0
765 \abovedisplayshortskip=\skip2
766 \belowdisplayskip=\skip4
767 \belowdisplayshortskip=\skip6\relax}
768 \def\widebaselines{\narrowbaselinesfalse\@currsize\selectfont}

```

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

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

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

```

769 \renewcommand{\normalsize}{%
770   \ifnarrowbaselines
771     \jsc@setfontsize\normalsize\@xpt\@xipt
772   \else
773     \jsc@setfontsize\normalsize\@xpt{\n@baseline}%
774   \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_EX 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、微調節してすることにしました。

```

775 \abovedisplayskip 11\jsc@mpt \@plus3\jsc@mpt \@minus4\jsc@mpt
776 \abovedisplayshortskip \z@ \@plus3\jsc@mpt
777 \belowdisplayskip 9\jsc@mpt \@plus3\jsc@mpt \@minus4\jsc@mpt
778 \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。

```

779 \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。

```

780 %% initialize
781 \normalsize

```

`\Cht` 基準となる長さの設定をします。pL^AT_EX 2_ε カーネル (`plfonts.dtx`) で宣言されている `\Cdp` 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 (`1zw`) です。

`\Cwd` [2017-08-31] 基準とする文字を「全角空白」（EUC コード 0xA1A1）から「漢」（JIS コード 0x3441）へ変更しました。

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

`\Chs` [X_YL^AT_EX-ja] 文字によってメトリクスが違うので決め打にします。

```

782 \newdimen\Cht
783 \newdimen\Cdp
784 \newdimen\Cwd
785 \newdimen\Cvs
786 \newdimen\Chs
787 \setlength\Cht{0.88\zw}
788 \setlength\Cdp{0.12\zw}
789 \setlength\Cwd{1\zw}
790 \setlength\Cvs{\baselineskip}
791 \setlength\Chs{1\zw}

```

\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 ± 2 , 2 ± 1 ポイントでしたが、ここではゼロ (\z@) にしました。

```

792 \newcommand{\small}{%
793   \ifnarrowbaselines
794     \!kiyou \jsc@setfontsize\small\@ixpt{11}%
795     \!kiyou \jsc@setfontsize\small{8.8888}{11}%
796   \else
797     \!kiyou \jsc@setfontsize\small\@ixpt{13}%
798     \!kiyou \jsc@setfontsize\small{8.8888}{13.2418}%
799   \fi
800   \abovedisplayskip 9\jsc@mpt \@plus3\jsc@mpt \@minus4\jsc@mpt
801   \abovedisplayshortskip \z@ \@plus3\jsc@mpt
802   \belowdisplayskip \abovedisplayskip
803   \belowdisplayshortskip \belowdisplayskip
804   \def\@listi{\leftmargin\leftmargin
805     \topsep \z@
806     \parsep \z@
807     \itemsep \parsep}}

```

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

```

808 \newcommand{\footnotesize}{%
809   \ifnarrowbaselines
810     \!kiyou \jsc@setfontsize\footnotesize\@viipt{9.5}%
811     \!kiyou \jsc@setfontsize\footnotesize{8.8888}{11}%
812   \else
813     \!kiyou \jsc@setfontsize\footnotesize\@viipt{11}%
814     \!kiyou \jsc@setfontsize\footnotesize{8.8888}{13.2418}%
815   \fi
816   \abovedisplayskip 6\jsc@mpt \@plus2\jsc@mpt \@minus3\jsc@mpt
817   \abovedisplayshortskip \z@ \@plus2\jsc@mpt
818   \belowdisplayskip \abovedisplayskip
819   \belowdisplayshortskip \belowdisplayskip
820   \def\@listi{\leftmargin\leftmargin
821     \topsep \z@

```

```

822          \parsep \z@
823          \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 824 \newcommand{\scriptsize}{\jsc@setfontsize\scriptsize\@viipt\@viipt}
825 \newcommand{\tiny}{\jsc@setfontsize\tiny\@vpt\@vpt}
\Huge 826 \if@twocolumn
827 \else
828 \fi
\HUGE 827 \langle\kiyou\rangle \newcommand{\large}{\jsc@setfontsize\large\@xiipt{\n@baseline}}
828 \langle\kiyou\rangle \newcommand{\large}{\jsc@setfontsize\large{11.111}{\n@baseline}}
829 \else
830 \langle\kiyou\rangle \newcommand{\large}{\jsc@setfontsize\large\@xiipt{17}}
831 \langle\kiyou\rangle \newcommand{\large}{\jsc@setfontsize\large{11.111}{17}}
832 \fi
833 \langle\kiyou\rangle \newcommand{\LARGE}{\jsc@setfontsize\LARGE\@xivpt{21}}
834 \langle\kiyou\rangle \newcommand{\LARGE}{\jsc@setfontsize\LARGE{12.222}{21}}
835 \newcommand{\LARGE}{\jsc@setfontsize\LARGE\@xivpt{25}}
836 \newcommand{\huge}{\jsc@setfontsize\huge\@xxpt{28}}
837 \newcommand{\Huge}{\jsc@setfontsize\Huge\@xxvpt{33}}
838 \newcommand{\HUGE}{\jsc@setfontsize\HUGE{30}{40}}

```

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

本文中の数式の中では `\narrowbaselines` 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。

```

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

```

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

見出し用のフォントは `\bfseries` 固定ではなく、`\headfont` という命令で定めることにします。これは太ゴシックが使えるときは `\sffamily \bfseries` でいいと思いますが、通常の中ゴシックでは単に `\sffamily` だけのほうがよさそうです。『`pLATEX 2ε` 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。

```

840 % \newcommand{\headfont}{\bfseries}
841 \newcommand{\headfont}{\gtfamily\sffamily}
842 % \newcommand{\headfont}{\sffamily\fontseries{sbc}\selectfont}

```

5 レイアウト

■二段組

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

```
843 <|kiyou>\setlength\columnsep{2\zw}  
844 <|kiyou>\setlength\columnsep{28truebp}  
845 \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。

```
\lineskiplimit 846 \setlength\lineskip{1\jsc@mp}  
847 \setlength\normallineskip{1\jsc@mp}  
\normallineskiplimit 848 \setlength\lineskiplimit{1\jsc@mp}  
849 \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。

```
850 \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。

```
851 \setlength\parskip{\z@}  
852 \if@slide  
853   \setlength\parindent{0\zw}  
854 \else  
855   \setlength\parindent{1\zw}  
856 \fi
```

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

```
\@highpenalty 857 \@lowpenalty 51  
858 \@medpenalty 151  
859 \@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。

```
860 % \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。

```
861 % \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 `\topskip` 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。

```
862 \setlength\topskip{1.38\zw}%% from 10\jsc@empt (2016-08-17)
863 \if@slide
864   \setlength\headheight{0\jsc@empt}
865 \else
866   \setlength\headheight{20\jsc@empt}%% from 2\topskip (2016-08-17); from \topskip (2003-06-26)
867 \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) となっていたのですが、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。

```
868 <*article | kiyou>
869 \if@slide
870   \setlength\footskip{0pt}
871 \else
872   \setlength\footskip{0.03367\paperheight}
873   \ifdim\footskip<\baselineskip
874     \setlength\footskip{\baselineskip}
875   \fi
876 \fi
877 </article | kiyou>
878 \jpf\setlength\footskip{9\jsc@mmm}
879 <*book>
880 \if@report
881   \setlength\footskip{0.03367\paperheight}
882   \ifdim\footskip<\baselineskip
883     \setlength\footskip{\baselineskip}
884   \fi
885 \else
```

```

886 \setlength\footskip{0pt}
887 \fi
888 \</book>
889 \<*report>
890 \setlength\footskip{0.03367\paperheight}
891 \ifdim\footskip<\baselineskip
892 \setlength\footskip{\baselineskip}
893 \fi
894 \</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)。

```

895 \<*article>
896 \if@slide
897 \setlength\headsep{0\jsc@empt}
898 \addtolength\headsep{-\topskip}%% added (2016-10-08)
899 \addtolength\headsep{10\jsc@empt}%% added (2016-10-08)
900 \else
901 \setlength\headsep{\footskip}
902 \addtolength\headsep{-\topskip}
903 \fi
904 \</article>
905 \<*book>
906 \if@report
907 \setlength\headsep{\footskip}
908 \addtolength\headsep{-\topskip}
909 \else
910 \setlength\headsep{6\jsc@mmm}
911 \addtolength\headsep{-\topskip}%% added (2016-10-08)
912 \addtolength\headsep{10\jsc@empt}%% added (2016-10-08)
913 \fi
914 \</book>
915 \<*report>
916 \setlength\headsep{\footskip}
917 \addtolength\headsep{-\topskip}
918 \</report>
919 \<*jspf>
920 \setlength\headsep{9\jsc@mmm}
921 \addtolength\headsep{-\topskip}
922 \</jspf>
923 \<*kiyou>
924 \setlength\headheight{0\jsc@empt}
925 \setlength\headsep{0\jsc@empt}
926 \addtolength\headsep{-\topskip}%% added (2016-10-08)
927 \addtolength\headsep{10\jsc@empt}%% added (2016-10-08)

```

928 \langle /kiyou \rangle

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

929 $\backslashsetlength\maxdepth\{.5\topskip\}$

■本文の幅と高さ

\backslashfullwidth 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 \backslashfullwidth という長さを定義します。

930 $\backslashnewdimen\fullwidth$

この \backslashfullwidth は article では紙幅 \backslashpaperwidth 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。

\backslashtextwidth 書籍以外では本文領域の幅 \backslashtextwidth は \backslashfullwidth 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 \times 2 段）+ 段間 8mm とします。

931 \langle *article \rangle

932 $\backslashif@slide$

933 $\backslashsetlength\fullwidth\{0.9\paperwidth\}$

934 \backslashelse

935 $\backslashsetlength\fullwidth\{0.76\paperwidth\}$

936 \backslashfi

937 $\backslashif@twocolumn \ @tempdima=2\zw \else \ @tempdima=1\zw \fi$

938 $\backslashdivide\fullwidth\ @tempdima \multiply\fullwidth\ @tempdima$

939 $\backslashsetlength\textwidth\{\fullwidth\}$

940 \langle /article \rangle

941 \langle *book \rangle

942 $\backslashif@report$

943 $\backslashsetlength\fullwidth\{0.76\paperwidth\}$

944 \backslashelse

945 $\backslashsetlength\fullwidth\{\paperwidth\}$

946 $\backslashaddtolength\fullwidth\{-36\jsc@mmm\}$

947 \backslashfi

948 $\backslashif@twocolumn \ @tempdima=2\zw \else \ @tempdima=1\zw \fi$

949 $\backslashdivide\fullwidth\ @tempdima \multiply\fullwidth\ @tempdima$

950 $\backslashsetlength\textwidth\{\fullwidth\}$

951 $\backslashif@report \else$

952 $\backslashif@twocolumn \else$

953 $\backslashifdim \fullwidth>40\zw$


```

954      \setlength\textwidth{40\zw}
955      \fi
956      \fi
957 \fi
958 \book
959 \report
960 \setlength\fullwidth{0.76\paperwidth}
961 \if@twocolumn \@tempdima=2\zw \else \@tempdima=1\zw \fi
962 \divide\fullwidth\@tempdima \multiply\fullwidth\@tempdima
963 \setlength\textwidth{\fullwidth}
964 \report
965 \jspf
966 \setlength\fullwidth{50\zw}
967 \addtolength\fullwidth{8\jsc@mmm}
968 \setlength\textwidth{\fullwidth}
969 \jspf
970 \kiyou
971 \setlength\fullwidth{48\zw}
972 \addtolength\fullwidth{\columnsep}
973 \setlength\textwidth{\fullwidth}
974 \kiyou

```

\textheight 紙の高さ \paperheight は、1 インチと \topmargin と \headheight と \headsep と \textheight と \footskip とページ下部の余白を加えたものです。

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

某学会誌スタイルでは 44 行にします。

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

[2016-08-26] \topskip を 10pt から 1.38zw に増やしたので、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)。

```

975 \article | book | report
976 \if@slide
977   \setlength{\textheight}{0.95\paperheight}
978 \else
979   \setlength{\textheight}{0.83\paperheight}
980 \fi
981 \addtolength{\textheight}{-10\jsc@empt}% from -\topskip (2016-10-08); from -\headheight (200
982 \addtolength{\textheight}{-\headsep}
983 \addtolength{\textheight}{-\footskip}
984 \addtolength{\textheight}{-\topskip}

```

```

985 \divide\textheight\baselineskip
986 \multiply\textheight\baselineskip
987 </article | book | report>
988 <jspf>\setlength{\textheight}{51\baselineskip}
989 <kiyou>\setlength{\textheight}{47\baselineskip}
990 \addtolength{\textheight}{\topskip}
991 \addtolength{\textheight}{0.1\jsc@empt}
992 <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 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$ での完全な `\flushbottom` の定義は

```

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

```

ですが、次のようにします。

```

993 \def\flushbottom{%
994   \def\@textbottom{\vskip \z@ \@plus.1\jsc@empt}%
995   \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。

```

996 \setlength\marginparsep{\columnsep}
997 \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。 $\text{T}_{\text{E}}\text{X}$ 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 $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$ (`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。

```

998 \setlength{\oddsidemargin}{\paperwidth}
999 \addtolength{\oddsidemargin}{-\fullwidth}
1000 \setlength{\oddsidemargin}{.5\oddsidemargin}
1001 \iftombow
1002   \addtolength{\oddsidemargin}{-1in}
1003 \else
1004   \addtolength{\oddsidemargin}{-\inv@mag in}
1005 \fi
1006 \setlength{\evensidemargin}{\oddsidemargin}
1007 \if@mparswitch
1008   \addtolength{\evensidemargin}{\fullwidth}
1009   \addtolength{\evensidemargin}{-\textwidth}
1010 \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。

```

1011 \setlength\marginparwidth{\paperwidth}

```

```

1012 \addtolength\marginparwidth{-\oddsidemargin}
1013 \addtolength\marginparwidth{-\inv@mag in}
1014 \addtolength\marginparwidth{-\textwidth}
1015 \addtolength\marginparwidth{-10\jsc@mmm}
1016 \addtolength\marginparwidth{-\marginparsep}
1017 \@tempdima=1\zw
1018 \divide\marginparwidth\@tempdima
1019 \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）。

```

1020 \setlength\topmargin{\paperheight}
1021 \addtolength\topmargin{-\textheight}
1022 \if@slide
1023   \addtolength\topmargin{-\headheight}
1024 \else
1025   \addtolength\topmargin{-10\jsc@mpt}%% from -\topskip (2016-10-08); from -\headheight (2003
1026 \fi
1027 \addtolength\topmargin{-\headsep}
1028 \addtolength\topmargin{-\footskip}
1029 \setlength\topmargin{0.5\topmargin}
1030 \kiyou\setlength\topmargin{81truebp}
1031 \iftombow
1032   \addtolength\topmargin{-1in}
1033 \else
1034   \addtolength\topmargin{-\inv@mag in}
1035 \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。

```

1036 {\footnotesize}\global\setlength\footnotesep{\baselineskip}
1037 \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。

```

1038 \setlength{\skip\footins}{16\jsc@mpt \@plus 5\jsc@mpt \@minus 2\jsc@mpt}

```

■フロート関連 フロート（図、表）関連のパラメータは L^AT_EX 2_ε 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。

1039 `\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。

1040 `\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。

1041 `\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。

1042 `\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。

1043 `\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。

1044 `\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。

1045 `\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。

1046 `\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。

1047 `\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。

1048 `\renewcommand{\dblfloatpagefraction}{.8}`

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

`\textfloatsep` 下部のフロートと本文との距離です。`\intextsep` は本文の途中に出力されるフロートと本

`\intextsep` 文との距離です。

1049 `\setlength\floatsep {12\jsc@empt \@plus 2\jsc@empt \@minus 2\jsc@empt}`

1050 `\setlength\textfloatsep{20\jsc@empt \@plus 2\jsc@empt \@minus 4\jsc@empt}`

1051 `\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 1052 \setlength\dblfloatsep {12\jsc@empt \@plus 2\jsc@empt \@minus 2\jsc@empt}
1053 \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 1054 \setlength\@fptop{0\jsc@empt \@plus 1fil}
1055 \setlength\@fpsep{8\jsc@empt \@plus 2fil}
1056 \setlength\@fpbot{0\jsc@empt \@plus 1fil}
```

`\@dblfpbot` 段抜きフロートについての値です。

```
\@dblfpsep 1057 \setlength\@dblfpbot{0\jsc@empt \@plus 1fil}
1058 \setlength\@dblfpsep{8\jsc@empt \@plus 2fil}
1059 \setlength\@dblfpbot{0\jsc@empt \@plus 1fil}
```

6 改ページ (日本語 T_EX 開発コミュニティ版のみ)

`\pltx@cleartorightpage` [2017-02-24] コミュニティ版 pL^AT_EX の標準クラス 2017/02/15 に合わせて, 同じ命令を追
`\pltx@cleartoleftpage` 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。

```
1060 <*article | book | report>
1061 \def\pltx@cleartorightpage{\clearpage\if@twoside
1062   \ifodd\c@page
1063     \IfDirectionTateT{%
1064       \hbox{}}\thispagestyle{empty}\newpage
1065     \if@twocolumn\hbox{}}\newpage\fi
1066   }%
1067 \else
1068   \IfDirectionYokoT{%
1069     \hbox{}}\thispagestyle{empty}\newpage
1070     \if@twocolumn\hbox{}}\newpage\fi
1071   }%
1072 \fi\fi}
1073 \def\pltx@cleartoleftpage{\clearpage\if@twoside
1074   \ifodd\c@page
1075     \IfDirectionYokoT{%
1076       \hbox{}}\thispagestyle{empty}\newpage
1077     \if@twocolumn\hbox{}}\newpage\fi
1078   }%
1079 \else
```

```

1080 \IfDirectionTateT{%
1081 \hbox{}\thispagestyle{empty}\newpage
1082 \if@twocolumn\hbox{}\newpage\fi
1083 }%
1084 \fi\fi}
1085 \def\pltx@cleartooddpage{\clearpage\if@twoside
1086 \ifodd\c@page\else
1087 \hbox{}\thispagestyle{empty}\newpage
1088 \if@twocolumn\hbox{}\newpage\fi
1089 \fi\fi}
1090 \def\pltx@cleartoevenpage{\clearpage\if@twoside
1091 \ifodd\c@page
1092 \hbox{}\thispagestyle{empty}\newpage
1093 \if@twocolumn\hbox{}\newpage\fi
1094 \fi\fi}
1095 </article | book | report>

```

`\cleardoublepage` [2017-02-24] コミュニティ版 p \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。

```

1096 <*book | report>
1097 \if@openleft
1098 \let\cleardoublepage\pltx@cleartoleftpage
1099 \else\if@openright
1100 \let\cleardoublepage\pltx@cleartorightpage
1101 \fi\fi
1102 </book | report>

```

7 ページスタイル

ページスタイルとして、 \LaTeX 2_ε (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 \LaTeX 2_ε 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ともに動作します。たとえば左マークを `\chapter`、右マークを `\section` で変更する場合はこれにあたります。しかし、同一ページに複数の `\markboth` があると、おかしい結果になることがあります。

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

`\ps@empty` `empty` ページスタイルの定義です。L^AT_EX 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。

```
1103 % \def\ps@empty{%
1104 %   \let\mkboth\gobbletwo
1105 %   \let\oddhead\empty
1106 %   \let\oddfoot\empty
1107 %   \let\evenhead\empty
1108 %   \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。

```
1109 \def\ps@plainfoot{%
1110   \let\mkboth\gobbletwo
1111   \let\oddhead\empty
1112   \def\oddfoot{\normalfont\hfil\thepage\hfil}%
1113   \let\evenhead\empty
1114   \let\evenfoot\oddfoot}
1115 \def\ps@plainhead{%
1116   \let\mkboth\gobbletwo
1117   \let\oddfoot\empty
1118   \let\evenfoot\empty
1119   \def\evenhead{%
1120     \if@mparswitch \hss \fi
1121     \hbox to \fullwidth{\textbf{\thepage}\hfil}%
1122     \if@mparswitch\else \hss \fi}%
1123   \def\oddhead{%
1124     \hbox to \fullwidth{\hfil\textbf{\thepage}\hss}}
1125 <book>\if@report \let\ps@plain\ps@plainfoot \else \let\ps@plain\ps@plainhead \fi
1126 <!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。

```
1127 <*article | kiyou>
1128 \if@twoside
1129   \def\ps@headings{%
1130     \let\oddfoot\empty
1131     \let\evenfoot\empty
```

```

1132 \def\@evenhead{\if@mparswitch \hss \fi
1133 \underline{\hbox to \fullwidth{\textbf{\thepage}\hfil\leftmark}}}%
1134 \if@mparswitch\else \hss \fi}%
1135 \def\@oddhead{%
1136 \underline{%
1137 \hbox to \fullwidth{\rightmark}\hfil\textbf{\thepage}}}\hss}%
1138 \let\@mkboth\markboth
1139 \def\sectionmark##1{\markboth{%
1140 \ifnum \c@secnumdepth >\z@ \thesection \hskip1\zw\fi
1141 ##1}}}%
1142 \def\subsectionmark##1{\markright{%
1143 \ifnum \c@secnumdepth >\@ne \thesubsection \hskip1\zw\fi
1144 ##1}}}%
1145 }
1146 \else % if not twoside
1147 \def\ps@headings{%
1148 \let\@oddfoot\@empty
1149 \def\@oddhead{%
1150 \underline{%
1151 \hbox to \fullwidth{\rightmark}\hfil\textbf{\thepage}}}\hss}%
1152 \let\@mkboth\markboth
1153 \def\sectionmark##1{\markright{%
1154 \ifnum \c@secnumdepth >\z@ \thesection \hskip1\zw\fi
1155 ##1}}}%
1156 \fi
1157 </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）。

```

1158 <*book | report>
1159 \newif\if@omit@number
1160 \def\ps@headings{%
1161 \let\@oddfoot\@empty
1162 \let\@evenfoot\@empty
1163 \def\@evenhead{%
1164 \if@mparswitch \hss \fi
1165 \underline{\hbox to \fullwidth{\autoxspacing
1166 \textbf{\thepage}\hfil\leftmark}}}%
1167 \if@mparswitch\else \hss \fi}%
1168 \def\@oddhead{\underline{\hbox to \fullwidth{\autoxspacing
1169 {\if@twoside\rightmark\else\leftmark\fi}\hfil\textbf{\thepage}}}\hss}%
1170 \let\@mkboth\markboth
1171 \def\chaptermark##1{\markboth{%
1172 \ifnum \c@secnumdepth >\m@ne
1173 <book> \if@mainmatter
1174 \if@omit@number\else
1175 \@chapapp\thechapter\@chappos\hskip1\zw
1176 \fi
1177 <book> \fi

```



```

1178     \fi
1179     ##1}{}}%
1180     \def\sectionmark##1{\markright{%
1181       \ifnum \c@secnumdepth >\z@ \thesection \hskip1\zw\fi
1182       ##1}}}%
1183 </book | report>

    最後は学会誌の場合です。

1184 <*jspf>
1185 \def\ps@headings{%
1186   \def\@oddfoot{\normalfont\hfil\thepage\hfil}
1187   \def\@evenfoot{\normalfont\hfil\thepage\hfil}
1188   \def\@oddhead{\normalfont\hfil \@title \hfil}
1189   \def\@evenhead{\normalfont\hfil プラズマ・核融合学会誌\hfil}
1190 </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。

```

1191 \def\ps@myheadings{%
1192   \let\@oddfoot\@empty\let\@evenfoot\@empty
1193   \def\@evenhead{%
1194     \if@mparswitch \hss \fi%
1195     \hbox to \fullwidth{\thepage\hfil\leftmark}%
1196     \if@mparswitch\else \hss \fi}%
1197   \def\@oddhead{%
1198     \hbox to \fullwidth{\rightmark\hfil\thepage}\hss}%
1199   \let\@mkboth\@gobbletwo
1200 <book | report> \let\chaptermark\@gobble
1201   \let\sectionmark\@gobble
1202 <|book&!report> \let\subsectionmark\@gobble
1203 }

```

8 文書のマークアップ

8.1 表題

`\title` これらは L^AT_EX 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 1204 % \newcommand*{\title}[1]{\gdef\@title{#1}}
      1205 % \newcommand*{\author}[1]{\gdef\@author{#1}}
      1206 % \newcommand*{\date}[1]{\gdef\@date{#1}}
      1207 % \date{\today}

```

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

```

\author 1208 <*jspf>
\keywords 1209 \newcommand*{\etitle}[1]{\gdef\@etitle{#1}}

```

```

1210 \newcommand*{\eauthor}[1]{\gdef\eauthor{#1}}
1211 \newcommand*{\keywords}[1]{\gdef\@keywords{#1}}
1212 \newcommand*{\email}[1]{\gdef\authors@mail{#1}}
1213 \newcommand*{\AuthorsEmail}[1]{\gdef\authors@mail{author's e-mail:\ #1}}
1214 \end{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なります。

```

1215 \def\plainifnotempty{%
1216   \ifx \@oddhead \@empty
1217     \ifx \@oddfoot \@empty
1218       \else
1219         \thispagestyle{plainfoot}%
1220       \fi
1221     \else
1222       \thispagestyle{plainhead}%
1223     \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。

```

1224 <*article | book | report | kiyou>
1225 \if@titlepage
1226   \newcommand{\maketitle}{%
1227     \begin{titlepage}%
1228       \let\footnotesize\small
1229       \let\footnoterule\relax
1230       \let\footnote\thanks
1231       \null\vfil
1232       \if@slide
1233         {\footnotesize \@date}%
1234       \begin{center}
1235         \mbox{} \ \ [1\zw]
1236         \large
1237         {\maybeblue\hrule height0\jsc@mpt depth2\jsc@mpt\relax}\par
1238         \jsc@smallskip
1239         \@title
1240         \jsc@smallskip
1241         {\maybeblue\hrule height0\jsc@mpt depth2\jsc@mpt\relax}\par
1242         \vfill
1243         {\small \@author}%
1244       \end{center}

```

```

1245     \else
1246     \vskip 60\jsc@mp
1247     \begin{center}%
1248         {\LARGE \@title \par}%
1249         \vskip 3em%
1250         {\large
1251             \lineskip .75em
1252             \begin{tabular}[t]{c}%
1253                 \@author
1254             \end{tabular}\par}%
1255         \vskip 1.5em
1256         {\large \@date \par}%
1257     \end{center}%
1258     \fi
1259     \par
1260     \@thanks\vfil\null
1261 \end{titlepage}%
1262 \setcounter{footnote}{0}%
1263 \global\let\thanks\relax
1264 \global\let\maketitle\relax
1265 \global\let\@thanks\@empty
1266 \global\let\@author\@empty
1267 \global\let\@date\@empty
1268 \global\let\@title\@empty
1269 \global\let\title\relax
1270 \global\let\author\relax
1271 \global\let\date\relax
1272 \global\let\and\relax
1273 }%
1274 \else
1275 \newcommand{\maketitle}{\par
1276     \begingroup
1277     \renewcommand\thefootnote{\@fnsymbol\c@footnote}%
1278     \def\@makefnmark{\rlap{\@textsuperscript{\normalfont\@thefnmark}}}%
1279     \long\def\@makefntext##1{\advance\leftskip 3\zw
1280         \parindent 1\zw\noindent
1281         \llap{\@textsuperscript{\normalfont\@thefnmark}\hskip0.3\zw}##1}%
1282     \if@twocolumn
1283         \ifnum \col@number=\@ne
1284             \@maketitle
1285         \else
1286             \twocolumn[\@maketitle]%
1287         \fi
1288     \else
1289         \newpage
1290         \global\@topnum\z@ % Prevents figures from going at top of page.
1291         \@maketitle
1292     \fi
1293     \plainifnotempty

```

```

1294     \@thanks
1295 \endgroup
1296 \setcounter{footnote}{0}%
1297 \global\let\thanks\relax
1298 \global\let\maketitle\relax
1299 \global\let\@thanks\@empty
1300 \global\let\@author\@empty
1301 \global\let\@date\@empty
1302 \global\let\@title\@empty
1303 \global\let\title\relax
1304 \global\let\author\relax
1305 \global\let\date\relax
1306 \global\let\and\relax
1307 }

```

`\@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。

```

1308 \def\@maketitle{%
1309   \newpage\null
1310   \vskip 2em
1311   \begin{center}%
1312     \let\footnote\thanks
1313     {\LARGE \@title \par}%
1314     \vskip 1.5em
1315     {\large
1316       \lineskip .5em
1317       \begin{tabular}[t]{c}%
1318         \@author
1319       \end{tabular}\par}%
1320     \vskip 1em
1321     {\large \@date}%
1322   \end{center}%
1323   \par\vskip 1.5em
1324   <article | report | kiyou> \ifvoid\@abstractbox\else\centerline{\box\@abstractbox}\vskip1.5em\fi
1325 }
1326 \fi
1327 </article | book | report | kiyou>
1328 <*jspf>
1329 \newcommand{\maketitle}{\par
1330   \begingroup
1331     \renewcommand\thefootnote{\@fnsymbol\c@footnote}%
1332     \def\@makefnmark{\rlap{\@textsuperscript{\normalfont\@thefnmark}}}%
1333     \long\def\@makefntext##1{\advance\leftskip 3\zw
1334       \parindent 1\zw\noindent
1335       \llap{\@textsuperscript{\normalfont\@thefnmark}\hskip0.3\zw}##1}%
1336     \twocolumn[\@maketitle]%
1337     \plainifnotempty
1338     \@thanks
1339   \endgroup
1340   \setcounter{footnote}{0}%

```

```

1341 \global\let\thanks\relax
1342 \global\let\maketitle\relax
1343 \global\let\@thanks\@empty
1344 \global\let\@author\@empty
1345 \global\let\@date\@empty
1346 % \global\let\@title\@empty % \@title は柱に使う
1347 \global\let\title\relax
1348 \global\let\author\relax
1349 \global\let\date\relax
1350 \global\let\and\relax
1351 \ifx\authors@mail\@undefined\else{%
1352     \def\@makefntext{\advance\leftskip 3\zw \parindent -3\zw}%
1353     \footnotetext[0]{\itshape\authors@mail}%
1354 } \fi
1355 \global\let\authors@mail\@undefined}
1356 \def\@maketitle{%
1357     \newpage\null
1358     \vskip 6em % used to be 2em
1359     \begin{center}
1360         \let\footnote\thanks
1361         \ifx\@title\@undefined\else{\LARGE\headfont\@title\par} \fi
1362         \lineskip .5em
1363         \ifx\@author\@undefined\else
1364             \vskip 1em
1365             \begin{tabular}[t]{c}%
1366                 \@author
1367             \end{tabular}\par
1368         \fi
1369         \ifx\@etitle\@undefined\else
1370             \vskip 1em
1371             {\large \@etitle \par}%
1372         \fi
1373         \ifx\@eauthor\@undefined\else
1374             \vskip 1em
1375             \begin{tabular}[t]{c}%
1376                 \@eauthor
1377             \end{tabular}\par
1378         \fi
1379         \vskip 1em
1380         \@date
1381     \end{center}
1382     \vskip 1.5em
1383     \centerline{\box\@abstractbox}
1384     \ifx\@keywords\@undefined\else
1385         \vskip 1.5em
1386         \centerline{\parbox{157\jsc@mmm}{\textsf{Keywords:}}\ \small\@keywords}}
1387     \fi
1388     \vskip 1.5em}
1389 \end{jspf}

```

8.2 章・節

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

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

それぞれの引数の意味は次の通りです。

名 ユーザレベルコマンドの名前です (例: section)。

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

字下げ 見出しの字下げ量です。

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

後アキ 正の場合は、見出しの下側の空きです。負の場合は、絶対値が見出しの右の空きです (見出しと同じ行から本文を始めます)。

スタイル 見出しの文字スタイルの設定です。

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

別見出し 目次や柱に出力する見出しです。

見出し 見出しです。

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

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

```
1390 \def\@startsection#1#2#3#4#5#6{%
1391   \if@noskipsec \leavevmode \fi
1392   \par
1393 % 見出し上の空きを \@tempskipa にセットする
1394   \@tempskipa #4\relax
1395 % \@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
1396   \if@english \@afterindentfalse \else \@afterindenttrue \fi
1397 % 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
1398   \ifdim \@tempskipa <\z@
1399     \@tempskipa -\@tempskipa \@afterindentfalse
1400   \fi
1401   \if@nobreak
1402 %   \everypar{\everyparhook}% これは間違い
1403     \everypar{}%
1404   \else
1405     \addpenalty\@secpenalty
1406 % 次の行は削除
```

```

1407 % \addvspace\@tempskipa
1408 % 次の \noindent まで追加
1409 \ifdim \@tempskipa >\z@
1410 \if@slide\else
1411 \null
1412 \vspace*{-\baselineskip}%
1413 \fi
1414 \vskip\@tempskipa
1415 \fi
1416 \fi
1417 \noindent
1418 % 追加終わり
1419 \@ifstar
1420 {\@ssect{#3}{#4}{#5}{#6}}%
1421 {\@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。

```

1422 \def\@sect#1#2#3#4#5#6[#7]#8{%
1423 \ifnum #2>\c@secnumdepth
1424 \let\@svsec\@empty
1425 \else
1426 \refstepcounter{#1}%
1427 \protected@edef\@svsec{\@secntformat{#1}\relax}%
1428 \fi
1429 % 見出し後の空きを \@tempskipa にセット
1430 \@tempskipa #5\relax
1431 % 条件判断の順序を入れ換えました
1432 \ifdim \@tempskipa<\z@
1433 \def\@svsechd{%
1434 #6{\hskip #3\relax
1435 \@svsec #8}%
1436 \csname #1mark\endcsname{#7}%
1437 \addcontentsline{toc}{#1}{%
1438 \ifnum #2>\c@secnumdepth \else
1439 \protect\numberline{\csname the#1\endcsname}%
1440 \fi
1441 #7}}% 目次にフルネームを載せるなら #8
1442 \else
1443 \begingroup
1444 \interlinepenalty \@M % 下から移動
1445 #6{%
1446 \@hangfrom{\hskip #3\relax\@svsec}%
1447 % \interlinepenalty \@M % 上に移動
1448 #8\@@par}%
1449 \endgroup
1450 \csname #1mark\endcsname{#7}%
1451 \addcontentsline{toc}{#1}{%
1452 \ifnum #2>\c@secnumdepth \else

```

```

1453      \protect\numberline{\csname the#1\endcsname}%
1454      \fi
1455      #7}% 目次にフルネームを載せるならここは #8
1456      \fi
1457      \@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。

```

1458 \def\@xsect#1{%
1459 % 見出しの後ろの空きを \@tempskipa にセット
1460   \@tempskipa #1\relax
1461 % 条件判断の順序を変えました
1462   \ifdim \@tempskipa<\z@
1463     \nobreakfalse
1464     \global\@noskipsecttrue
1465     \everypar{%
1466       \if@noskipsec
1467         \global\@noskipsecfalse
1468         {\setbox\z@\lastbox}%
1469         \clubpenalty\@M
1470         \begingroup \@svsechd \endgroup
1471         \unskip
1472         \@tempskipa #1\relax
1473         \hskip -\@tempskipa
1474       \else
1475         \clubpenalty \@clubpenalty
1476       \everypar{\everyparhook}%
1477     \fi
1478 %     \everyparhook
1479   }%
1480 \else
1481   \par \nobreak
1482   \vskip \@tempskipa
1483   \@afterheading
1484 \fi
1485 \if@slide
1486   {\vskip\if@twocolumn-5\jsc@mpt\else-6\jsc@mpt\fi
1487    \maybeblue\hrule height0\jsc@mpt depth1\jsc@mpt
1488    \vskip\if@twocolumn 4\jsc@mpt\else 7\jsc@mpt\fi\relax}%
1489 \fi
1490 \par % 2000-12-18
1491 \ignorespaces}
1492 \def\@ssect#1#2#3#4#5{%
1493   \@tempskipa #3\relax
1494   \ifdim \@tempskipa<\z@
1495     \def\@svsechd{#4{\hskip #1\relax #5}}%

```



```

1496 \else
1497 \begingroup
1498 #4{%
1499 \@hangfrom{\hskip #1}%
1500 \interlinepenalty \@M #5\@@par}%
1501 \endgroup
1502 \fi
1503 \@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^AT_EX 本体で定義済みです。

```

\subsectionmark 1504 \newcommand*\chaptermark[1]{}
\subsubsectionmark 1505 % \newcommand*\sectionmark[1]{}
1506 % \newcommand*\subsectionmark[1]{}
\paragraphmark 1507 % \newcommand*\subsubsectionmark[1]{}
\subparagraphmark 1508 % \newcommand*\paragraphmark[1]{}
1509 % \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決めるカウンタです。

```

1510 <|book&!report>\setcounter{secnumdepth}{3}
1511 <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 1512 \newcounter{part}
1513 <book | report>\newcounter{chapter}
\c@subsubsection 1514 <book | report>\newcounter{section}[chapter]
1515 <|book&!report>\newcounter{section}
\c@paragraph 1516 \newcounter{subsection}[section]
1517 \newcounter{subsubsection}[subsection]
1518 \newcounter{paragraph}[subsubsection]
1519 \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。

```
1520 \renewcommand{\thepart}{\@Roman\c@part}
1521 \langle!book&!report\rangle% \renewcommand{\thesection}{\@arabic\c@section}
1522 \langle!book&!report\rangle\renewcommand{\thesection}{\presectionname\@arabic\c@section\postsectionname}
1523 \langle!book&!report\rangle\renewcommand{\thesubsection}{\@arabic\c@section.\@arabic\c@subsection}
1524 \langle*book | report\rangle
1525 \renewcommand{\thechapter}{\@arabic\c@chapter}
1526 \renewcommand{\thesection}{\thechapter.\@arabic\c@section}
1527 \renewcommand{\thesubsection}{\thesection.\@arabic\c@subsection}
1528 \langle/book | report\rangle
1529 \renewcommand{\thesubsubsection}{\%
1530 \thesubsection.\@arabic\c@subsubsection}
1531 \renewcommand{\theparagraph}{\%
1532 \thesubsubsection.\@arabic\c@paragraph}
1533 \renewcommand{\thesubparagraph}{\%
1534 \theparagraph.\@arabic\c@subparagraph}
```

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

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

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

[2003-03-02] `\@secapp` は外しました。

```
1535 \langle book | report \rangle \newcommand{\@chapapp}{\prechaptername}
1536 \langle book | report \rangle \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` かどうかに関らず奇数ページまで繰るように修正することで、問題を解消しました。実は、 \LaTeX 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 \LaTeX の標準クラス 2017/03/05 も参照)。

```
1537 \langle *book \rangle
1538 \newcommand\frontmatter{\%
1539 \pltx@cleartooddpage
1540 \@mainmatterfalse
1541 \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。

```
1542 \newcommand\mainmatter{\%
1543 \pltx@cleartooddpage
1544 \@mainmattertrue
1545 \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。

```

1546 \newcommand\backmatter{%
1547   \if@openleft
1548     \cleardoublepage
1549   \else\if@openright
1550     \cleardoublepage
1551   \else
1552     \clearpage
1553   \fi\fi
1554   \@mainmatterfalse}
1555 \</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 \CMDDB }
\def\CMDA      [#1]#2{...} % \chapter[...]{...} の定義
\def\CMDDB     #1{...}     % \chapter*{...} の定義

```

まず `book` と `report` のクラス以外です。

```

1556 \<!*book&!report>
1557 \newcommand\part{%
1558   \if@noskipsec \leavevmode \fi
1559   \par
1560   \addvspace{4ex}%
1561   \if@english \@afterindentfalse \else \@afterindenttrue \fi
1562   \secdef\@part\@spart}
1563 \</!book&!report>

```

`book` および `report` クラスの場合は、少し複雑です。

```

1564 \<*book | report>
1565 \newcommand\part{%
1566   \if@openleft
1567     \cleardoublepage
1568   \else\if@openright
1569     \cleardoublepage
1570   \else
1571     \clearpage
1572   \fi\fi
1573   \thispagestyle{empty}% 欧文用標準スタイルでは plain
1574   \if@twocolumn
1575     \onecolumn

```

```

1576 \restonecoltrue
1577 \else
1578 \restonecolfalse
1579 \fi
1580 \null\vfil
1581 \secdef\@part\@spart}
1582 </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。

```

1583 < *!book&!report>
1584 \def\@part[#1]#2{%
1585 \ifnum \c@secnumdepth >\m@ne
1586 \refstepcounter{part}%
1587 \addcontentsline{toc}{part}{%
1588 \prepartname\thepart\postpartname\hspace{1\zw}#1}%
1589 \else
1590 \addcontentsline{toc}{part}{#1}%
1591 \fi
1592 \markboth{}{}%
1593 {\parindent\z@
1594 \raggedright
1595 \interlinepenalty \@M
1596 \normalfont
1597 \ifnum \c@secnumdepth >\m@ne
1598 \Large\headfont\prepartname\thepart\postpartname
1599 \par\nobreak
1600 \fi
1601 \huge \headfont #2%
1602 \markboth{}{}\par}%
1603 \nobreak
1604 \vskip 3ex
1605 \@afterheading}
1606 < /!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。

```

1607 < *book | report>
1608 \def\@part[#1]#2{%
1609 \ifnum \c@secnumdepth >-2\relax
1610 \refstepcounter{part}%
1611 \addcontentsline{toc}{part}{%
1612 \prepartname\thepart\postpartname\hspace{1\zw}#1}%
1613 \else
1614 \addcontentsline{toc}{part}{#1}%
1615 \fi
1616 \markboth{}{}%
1617 {\centering
1618 \interlinepenalty \@M

```

```

1619 \normalfont
1620 \ifnum \c@secnumdepth >-2\relax
1621 \huge\headfont \prepartname\thepart\postpartname
1622 \par\vskip20\jsc@mp
1623 \fi
1624 \Huge \headfont #2\par}%
1625 \@endpart}
1626 </book | report>

```

`\@spart` 番号を付けない部です。

```

1627 <*!book&!report>
1628 \def\@spart#1{%
1629 \parindent \z@ \raggedright
1630 \interlinepenalty \@M
1631 \normalfont
1632 \huge \headfont #1\par}%
1633 \nobreak
1634 \vskip 3ex
1635 \@afterheading}
1636 </*!book&!report>
1637 <*book | report>
1638 \def\@spart#1{%
1639 \centering
1640 \interlinepenalty \@M
1641 \normalfont
1642 \Huge \headfont #1\par}%
1643 \@endpart}
1644 </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 \LaTeX では `classes.dtx` v1.4b (2000/05/19) で修正されています。

```

1645 <*book | report>
1646 \def\@endpart{\vfil\newpage
1647 \if@twoside
1648 \if@openleft %% added (2017/02/24)
1649 \null\thispagestyle{empty}\newpage
1650 \else\if@openright %% added (2016/12/13)
1651 \null\thispagestyle{empty}\newpage
1652 \fi\fi %% added (2016/12/13, 2017/02/24)
1653 \fi
1654 \if@restonecol
1655 \twocolumn
1656 \fi}
1657 </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。

```

1658 <*book | report>
1659 \newcommand{\chapter}{%
1660   \if@openleft\cleardoublepage\else
1661   \if@openright\cleardoublepage\else\clearpage\fi\fi
1662   \plainifnotempty % 元: \thispagestyle{plain}
1663   \global\@topnum\z@
1664   \if@english \@afterindentfalse \else \@afterindenttrue \fi
1665   \secdef
1666     {\@omit@numberfalse\@chapter}%
1667     {\@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。

```

1668 \def\@chapter[#1]#2{%
1669   \ifnum \c@secnumdepth >\m@ne
1670 <book>   \if@mainmatter
1671           \refstepcounter{chapter}%
1672           \typeout{\@chapapp\thechapter\@chappos}%
1673           \addcontentsline{toc}{chapter}%
1674             {\protect\numberline
1675              % {\if@english\thechapter\else\@chapapp\thechapter\@chappos\fi}%
1676              {\@chapapp\thechapter\@chappos}%
1677              #1}%
1678 <book>   \else\addcontentsline{toc}{chapter}{#1}\fi
1679   \else
1680     \addcontentsline{toc}{chapter}{#1}%
1681   \fi
1682   \chaptermark{#1}%
1683   \addtocontents{lof}{\protect\addvspace{10\jsc@empt}}%
1684   \addtocontents{lot}{\protect\addvspace{10\jsc@empt}}%
1685   \if@twocolumn
1686     \@topnewpage[\@makechapterhead{#2}]%
1687   \else
1688     \@makechapterhead{#2}%
1689     \@afterheading
1690   \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。

```

1691 \def\@makechapterhead#1{%
1692   \vspace*{2\Cvs}% 欧文は 50pt
1693   {\parindent \z@ \raggedright \normalfont
1694     \ifnum \c@secnumdepth >\m@ne
1695 <book>   \if@mainmatter
1696           \huge\headfont \@chapapp\thechapter\@chappos

```

```

1697      \par\nobreak
1698      \vskip \Cvs % 欧文は 20pt
1699 <book>      \fi
1700      \fi
1701      \interlinepenalty\@M
1702      \Huge \headfont #1\par\nobreak
1703      \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。

```

1704 \def\@schapter#1{%
1705   \chaptermark{#1}%
1706   \if@twocolumn
1707     \topnewpage[\@makeschapterhead{#1}]%
1708   \else
1709     \@makeschapterhead{#1}\@afterheading
1710   \fi}

```

`\@makeschapterhead` 番号なしの章見出しです。

```

1711 \def\@makeschapterhead#1{%
1712   \vspace*{2\Cvs}% 欧文は 50pt
1713   {\parindent \z@ \raggedright
1714     \normalfont
1715     \interlinepenalty\@M
1716     \Huge \headfont #1\par\nobreak
1717     \vskip 3\Cvs}} % 欧文は 40pt
1718 </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。

```

1719 \if@twocolumn
1720   \newcommand{\section}{%
1721     <jspf>\ifx\maketitle\relax\else\maketitle\fi
1722     \@startsection{section}{1}{\z@}%
1723     <!kiyou> {0.6\Cvs}{0.4\Cvs}%
1724     <kiyou>  {\Cvs}{0.5\Cvs}%
1725     % {\normalfont\large\headfont\@secapp}}
1726     {\normalfont\large\headfont\raggedright}}
1727 \else
1728   \newcommand{\section}{%
1729     \if@slide\clearpage\fi
1730     \@startsection{section}{1}{\z@}%
1731     {\Cvs \@plus.5\Cdp \@minus.2\Cdp}% 前アキ
1732     {.5\Cvs \@plus.3\Cdp}% 後アキ
1733     % {\normalfont\Large\headfont\@secapp}}
1734     {\normalfont\Large\headfont\raggedright}}

```

1735 \fi

\subsection 同上です。

```
1736 \if@twocolumn
1737   \newcommand{\subsection}{\@startsection{subsection}{2}{\z@}%
1738     {\z@}{\if@slide .4\Cvs \else \z@ \fi}%
1739     {\normalfont\normalsize\headfont}}
1740 \else
1741   \newcommand{\subsection}{\@startsection{subsection}{2}{\z@}%
1742     {\Cvs \@plus.5\Cdp \@minus.2\Cdp}% 前アキ
1743     {.5\Cvs \@plus.3\Cdp}% 後アキ
1744     {\normalfont\large\headfont}}
1745 \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)。

```
1746 \if@twocolumn
1747   \newcommand{\subsubsection}{\@startsection{subsubsection}{3}{\z@}%
1748     {\z@}{\if@slide .4\Cvs \else \z@ \fi}%
1749     {\normalfont\normalsize\headfont}}
1750 \else
1751   \newcommand{\subsubsection}{\@startsection{subsubsection}{3}{\z@}%
1752     {\Cvs \@plus.5\Cdp \@minus.2\Cdp}%
1753     {\if@slide .5\Cvs \@plus.3\Cdp \else \z@ \fi}%
1754     {\normalfont\normalsize\headfont}}
1755 \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。

```
1756 \!jspf\newcommand{\jsParagraphMark}{■}
1757 \if@twocolumn
1758   \newcommand{\paragraph}{\@startsection{paragraph}{4}{\z@}%
1759     {\z@}{\if@slide .4\Cvs \else -1\zw\fi}% 改行せず 1zw のアキ
1760 \!jspf} {\normalfont\normalsize\headfont}}
1761 \!jspf} {\normalfont\normalsize\headfont\jsParagraphMark}}
1762 \else
1763   \newcommand{\paragraph}{\@startsection{paragraph}{4}{\z@}%
1764     {0.5\Cvs \@plus.5\Cdp \@minus.2\Cdp}%
1765     {\if@slide .5\Cvs \@plus.3\Cdp \else -1\zw\fi}% 改行せず 1zw のアキ
1766 \!jspf} {\normalfont\normalsize\headfont}}
1767 \!jspf} {\normalfont\normalsize\headfont\jsParagraphMark}}
1768 \fi
```


`\subparagraph` 見出しの後ろで改行されません。

```
1769 \if@twocolumn
1770   \newcommand{\subparagraph}{\@startsection{subparagraph}{5}{\z@}%
1771     {\z@}{\if@slide .4\Cvs \@plus.3\Cdp \else -1\zw\fi}%
1772     {\normalfont\normalsize\headfont}}
1773 \else
1774   \newcommand{\subparagraph}{\@startsection{subparagraph}{5}{\z@}%
1775     {\z@}{\if@slide .5\Cvs \@plus.3\Cdp \else -1\zw\fi}%
1776     {\normalfont\normalsize\headfont}}
1777 \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 `\leftmargin k` 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。

```
1778 \if@slide
1779   \setlength\leftmargini{1\zw}
1780 \else
1781   \if@twocolumn
1782     \setlength\leftmargini{2\zw}
1783   \else
1784     \setlength\leftmargini{3\zw}
1785   \fi
1786 \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 `\leftmarginiii` なっています。ここでは全角幅の整数倍に丸めました。

```
\leftmarginiv 1787 \if@slide
\leftmarginv 1788   \setlength\leftmarginii {1\zw}
1789   \setlength\leftmarginiii {1\zw}
\leftmarginvi 1790   \setlength\leftmarginiv {1\zw}
1791   \setlength\leftmarginv {1\zw}
1792   \setlength\leftmarginvi {1\zw}
1793 \else
1794   \setlength\leftmarginii {2\zw}
1795   \setlength\leftmarginiii {2\zw}
1796   \setlength\leftmarginiv {2\zw}
1797   \setlength\leftmarginv {1\zw}
1798   \setlength\leftmarginvi {1\zw}
1799 \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。

```
1800 \setlength \labelsep {0.5\zw} % .5em
1801 \setlength \labelwidth{\leftmargini}
1802 \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。

```
1803 \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 1804 \@beginparpenalty -\@lowpenalty
\@itempenalty 1805 \@endparpenalty -\@lowpenalty
1806 \@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。

```
1807 \def\@listi{\leftmargin\leftmargini
1808 \parsep \z@
1809 \topsep 0.5\baselineskip
1810 \itemsep \z@ \relax}
1811 \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）。

```
1812 \@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 1813 \def\@listii{\leftmargin\leftmarginii
\@listiv 1814 \labelwidth\leftmarginii \advance\labelwidth-\labelsep
1815 \topsep \z@
\@listv 1816 \parsep \z@
\@listvi 1817 \itemsep\parsep}
1818 \def\@listiii{\leftmargin\leftmarginiii
1819 \labelwidth\leftmarginiii \advance\labelwidth-\labelsep
1820 \topsep \z@
1821 \parsep \z@
1822 \itemsep\parsep}
1823 \def\@listiv {\leftmargin\leftmarginiv
1824 \labelwidth\leftmarginiv
1825 \advance\labelwidth-\labelsep}
1826 \def\@listv {\leftmargin\leftmarginv
1827 \labelwidth\leftmarginv
```

```

1828             \advance\labelwidth-\labelsep}
1829 \def\@listvi {\leftmargin\leftmarginvi
1830             \labelwidth\leftmarginvi
1831             \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^AT_EX 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。

```

1832 \renewcommand{\theenumi}{\@arabic\c@enumi}
1833 \renewcommand{\theenumii}{\@alph\c@enumii}
1834 \renewcommand{\theenumiii}{\@roman\c@enumiii}
1835 \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 1836 \newcommand{\labelenumi}{\theenumi.}
1837 \newcommand{\labelenumii}{\inhibitglue (\theenumii ) \inhibitglue}
1838 \newcommand{\labelenumiii}{\theenumiii.}
1839 \newcommand{\labelenumiv}{\theenumiv.}

```

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

```

\p@enumiv 1840 \renewcommand{\p@enumii}{\theenumi}
1841 \renewcommand{\p@enumiii}{\theenumi\inhibitglue (\theenumii ) }
1842 \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 1843 \newcommand\labelitemi{\textbullet}
1844 \newcommand\labelitemii{\normalfont\bfseries \textendash}
\labelitemiii 1845 \newcommand\labelitemiii{\textasteriskcentered}
\labelitemiv 1846 \newcommand\labelitemiv{\textperiodcentered}

```

■description 環境

`description (env.)` 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。

```

1847 \newenvironment{description}{%
1848   \list{}{%
1849     \labelwidth=\leftmargin

```

```

1850 \labelsep=1\zw
1851 \advance \labelwidth by -\labelsep
1852 \let \makelabel=\descriptionlabel}}{\endlist}

```

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

```

1853 \newcommand*\descriptionlabel[1]{\normalfont\headfont #1\hfil}

```

■概要

`abstract` (*env.*) 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。

```

1854 <*book>
1855 \newenvironment{abstract}{%
1856 \begin{list}{}{%
1857 \listparindent=1\zw
1858 \itemindent=\listparindent
1859 \rightmargin=0pt
1860 \leftmargin=5\zw}\item[]}{\end{list}\vspace{\baselineskip}}
1861 </book>
1862 <*article | report | kiyou>
1863 \newbox\@abstractbox
1864 \if@titlepage
1865 \newenvironment{abstract}{%
1866 \titlepage
1867 \null\vfil
1868 \@beginparpenalty\@lowpenalty
1869 \begin{center}%
1870 \headfont \abstractname
1871 \@endparpenalty\@M
1872 \end{center}}%
1873 {\par\vfil\null\endtitlepage}
1874 \else
1875 \newenvironment{abstract}{%
1876 \if@twocolumn
1877 \ifx\maketitle\relax
1878 \section*{\abstractname}%
1879 \else
1880 \global\setbox\@abstractbox\hbox\bgroup
1881 \begin{minipage}[b]{\textwidth}
1882 \small\parindent1\zw
1883 \begin{center}%
1884 {\headfont \abstractname\vspace{-.5em}\vspace{\z@}}%
1885 \end{center}%

```

```

1886         \list{}{%
1887             \listparindent\parindent
1888             \itemindent \listparindent
1889             \rightmargin \leftmargin}%
1890         \item\relax
1891     \fi
1892 \else
1893     \small
1894     \begin{center}%
1895         {\headfont \abstractname\vspace{-.5em}\vspace{\z@}}%
1896     \end{center}%
1897     \list{}{%
1898         \listparindent\parindent
1899         \itemindent \listparindent
1900         \rightmargin \leftmargin}%
1901     \item\relax
1902 \fi}{\if@twocolumn
1903     \ifx\maketitle\relax
1904     \else
1905         \endlist\end{minipage}\egroup
1906     \fi
1907 \else
1908     \endlist
1909 \fi}
1910 \fi
1911 </article | report | kiyou>
1912 <*jspf>
1913 \newbox\@abstractbox
1914 \newenvironment{abstract}{%
1915     \global\setbox\@abstractbox\hbox\bgroup
1916     \begin{minipage}[b]{157\jsc@mmm}{\sffamily Abstract}\par
1917     \small
1918     \if@english \parindent6\jsc@mmm \else \parindent1\zw \fi}%
1919     {\end{minipage}\egroup}
1920 </jspf>

```

■キーワード

keywords (env.) 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。

```

1921 <*jspf>
1922 %\newbox\@keywordsbox
1923 %\newenvironment{keywords}{%
1924 %     \global\setbox\@keywordsbox\hbox\bgroup
1925 %     \begin{minipage}[b]{1570\jsc@mmm}{\sffamily Keywords:}\par
1926 %         \small\parindent0\zw}%
1927 %     {\end{minipage}\egroup}
1928 </jspf>

```

■verse 環境

`verse (env.)` 詩のための `verse` 環境です。

```
1929 \newenvironment{verse}{%
1930   \let \=\@centercr
1931   \list{}{%
1932     \itemsep \z@
1933     \itemindent -2\zw % 元: -1.5em
1934     \listparindent\itemindent
1935     \rightmargin \z@
1936     \advance\leftmargin 2\zw}% 元: 1.5em
1937   \item\relax}{\endlist}
```

■quotation 環境

`quotation (env.)` 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。

```
1938 \newenvironment{quotation}{%
1939   \list{}{%
1940     \listparindent\parindent
1941     \itemindent\listparindent
1942     \rightmargin \z@}%
1943   \item\relax}{\endlist}
```

■quote 環境

`quote (env.)` `quote` 環境は、段落がインデントされないことを除き、`quotation` 環境と同じです。

```
1944 \newenvironment{quote}%
1945   {\list{}{\rightmargin\z@}\item\relax}{\endlist}
```

■定理など `ltthm.dtx` 参照。たとえば次のように定義します。

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

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

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

```
1946 \def\@begintheorem#1#2{\trivlist\labelsep=1\zw
1947   \item[\hskip \labelsep{\headfont #1\ #2}]}
1948 \def\@opargbegintheorem#1#2#3{\trivlist\labelsep=1\zw
1949   \item[\hskip \labelsep{\headfont #1\ #2 (#3)}]}
```

`titlepage (env.)` 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^AT_EX 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^AT_EX の標準クラス 2017/02/15 に合わせましたが、こちらも片面印刷あるいは独立のタイトルページを作らないクラスばかりでしたので、従来の挙動は何も変わらずに済みました。

```

1950 \newenvironment{titlepage}{%
1951 \book} \pltx@cleartooddpage %% 2017-02-24
1952 \if@twocolumn
1953 \@restonecoltrue\onecolumn
1954 \else
1955 \@restonecolfalse\newpage
1956 \fi
1957 \thispagestyle{empty}%
1958 \ifodd\c@page\setcounter{page}\@ne\else\setcounter{page}\z@\fi %% 2017-02-24
1959 }%
1960 {\if@restonecol\twocolumn \else \newpage \fi
1961 \if@twoside\else
1962 \setcounter{page}\@ne
1963 \fi}

```

■付録

\appendix 本文と付録を分離するコマンドです。

```

1964 <*\book&!report>
1965 \newcommand{\appendix}{\par
1966 \setcounter{section}{0}%
1967 \setcounter{subsection}{0}%
1968 \gdef\presectionname{\appendixname}%
1969 \gdef\postsectionname{}}%
1970 % \gdef\thesection{\@Alph\c@section}% [2003-03-02]
1971 \gdef\thesection{\presectionname\@Alph\c@section\postsectionname}%
1972 \gdef\thesubsection{\@Alph\c@section.\@arabic\c@subsection}}
1973 </!\book&!report>
1974 <*\book | report>
1975 \newcommand{\appendix}{\par
1976 \setcounter{chapter}{0}%
1977 \setcounter{section}{0}%
1978 \gdef\@chapapp{\appendixname}%
1979 \gdef\@chappos{}}%
1980 \gdef\thechapter{\@Alph\c@chapter}}
1981 </\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 倍の幅の空が入ります。

```
1982 \setlength\arraycolsep{5\jsc@empt}
```

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

```
1983 \setlength\tabcolsep{6\jsc@empt}
```

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

```
1984 \setlength\arrayrulewidth{.4\jsc@empt}
```

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

```
1985 \setlength\doublerulesep{2\jsc@empt}
```

■tabbing 環境

`\tabbingsep` \ ' コマンドで入るアキです。

```
1986 \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。

```
1987 \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です。

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

```
1988 \setlength\fboxsep{3\jsc@empt}
```

```
1989 \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。

```
1990 <!\book&!report>\renewcommand \theequation {\@arabic\c@equation}
```

```
1991 <*\book | report>
```

```
1992 \addtoreset{equation}{chapter}
```

```
1993 \renewcommand\theequation
```

```
1994 {\ifnum \c@chapter>\z@ \thechapter.\fi \@arabic\c@equation}
```

```
1995 </\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。

```
1996 % \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。

```
1997 % \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。

```
1998 % \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。

```
1999 <!*book&!report>
2000 \newcounter{figure}
2001 \renewcommand \thefigure {\@arabic\c@figure}
2002 <\/!book&!report>
2003 <*book | report>
2004 \newcounter{figure}[chapter]
2005 \renewcommand \thefigure
2006     {\ifnum \c@chapter>\z@ \thechapter.\fi \@arabic\c@figure}
2007 <\/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 2008 \def\fps@figure{tbp}
\ftype@figure 2009 \def\ftype@figure{1}
2010 \def\ext@figure{lof}
2011 \def\fnum@figure{\figurename\nobreak\thefigure}
```

`figure (env.)` * 形式は段抜きのフロートです。

```
figure* (env.) 2012 \newenvironment{figure}%
2013     {\@float{figure}}%
2014     {\end@float}
2015 \newenvironment{figure*}%
```

```
2016          {\@dblfloat{figure}}}%
2017          {\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。

```
2018 < *!book&!report>
2019 \newcounter{table}
2020 \renewcommand\thetable{\@arabic\c@table}
2021 < /!book&!report>
2022 < *book | report>
2023 \newcounter{table}[chapter]
2024 \renewcommand \thetable
2025     {\ifnum \c@chapter>\z@ \thechapter.\fi \@arabic\c@table}
2026 < /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。
`\ftype@table` た。

```
\ext@table 2027 \def\fps@table{tbp}
2028 \def\ftype@table{2}
\fnun@table 2029 \def\ext@table{lot}
2030 \def\fnun@table{\tablename\nobreak\thetable}
```

`table (env.)` * は段抜きのフロートです。

```
table* (env.) 2031 \newenvironment{table}%
2032             {\@float{table}}%
2033             {\end@float}
2034 \newenvironment{table*}%
2035             {\@dblfloat{table}}%
2036             {\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ふのを直しました。

```
2037 \newlength\abovecaptionskip
2038 \newlength\belowcaptionskip
2039 \setlength\abovecaptionskip{5\jsc@empt} % 元: 10\p@
2040 \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。

```
2041 <!*jspf>
2042 % \long\def\@makecaption#1#2{\small
2043 %   \advance\leftskip10\jsc@mmm
2044 %   \advance\rightskip10\jsc@mmm
2045 %   \vskip\abovcaptionskip
2046 %   \sbox\@tempboxa{#1\hskip1\zw\relax #2}%
2047 %   \ifdim \wd\@tempboxa >\hsize
2048 %     #1\hskip1\zw\relax #2\par
2049 %   \else
2050 %     \global \@minipagefalse
2051 %     \hb@xt@\hsize{\hfil\box\@tempboxa\hfil}%
2052 %   \fi
2053 %   \vskip\belowcaptionskip}}
2054 \long\def\@makecaption#1#2{\small
2055   \advance\leftskip .0628\linewidth
2056   \advance\rightskip .0628\linewidth
2057   \vskip\abovcaptionskip
2058   \sbox\@tempboxa{#1\hskip1\zw\relax #2}%
2059   \ifdim \wd\@tempboxa <\hsize \centering \fi
2060   #1{\hskip1\zw\relax #2}\par
2061   \vskip\belowcaptionskip}}
2062 </!jspf>
2063 <!*jspf>
2064 \long\def\@makecaption#1#2{%
2065   \vskip\abovcaptionskip
2066   \sbox\@tempboxa{\small\sffamily #1\quad #2}%
2067   \ifdim \wd\@tempboxa >\hsize
2068     {\small\sffamily
2069       \list{#1}{%
2070         \renewcommand{\makelabel}[1]{##1\hfil}
2071         \itemsep \z@
2072         \itemindent \z@
2073         \labelsep \z@
2074         \labelwidth 11\jsc@mmm
2075         \listparindent\z@
2076         \leftmargin 11\jsc@mmm}\item\relax #2\endlist}
2077   \else
2078     \global \@minipagefalse
2079     \hb@xt@\hsize{\hfil\box\@tempboxa\hfil}%
2080   \fi
2081   \vskip\belowcaptionskip}
2082 </jspf>
```

9 フォントコマンド

ここでは L^AT_EX 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 2083 \DeclareOldFontCommand{\mc}{\normalfont\mcfamily}{\mathmc}
\rm 2084 \DeclareOldFontCommand{\gt}{\normalfont\gtfamily}{\mathgt}
\rm 2085 \DeclareOldFontCommand{\rm}{\normalfont\rmfamily}{\mathrm}
\sff 2086 \DeclareOldFontCommand{\sf}{\normalfont\sffamily}{\mathsf}
\tt 2087 \DeclareOldFontCommand{\tt}{\normalfont\ttfamily}{\mathtt}
```

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

```
2088 \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ップスは数式中では何もしま
`\sl` せん（警告メッセージを出力します）。通常のアップライト体に戻るコマンドは `\upshape`
`\sc` です。

```
2089 \DeclareOldFontCommand{\it}{\normalfont\itshape}{\mathit}
2090 \DeclareOldFontCommand{\sl}{\normalfont\slshape}{\@nomath\sl}
2091 \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 2092 \DeclareRobustCommand*{\cal}{\@fontswitch\relax\mathcal}
2093 \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。

```
2094 \newcommand\@pnumwidth{1.55em}
2095 \newcommand\@tocrmarg{2.55em}
2096 \newcommand\@dotsep{4.5}
2097 <!book&!report>\setcounter{tocdepth}{2}
2098 <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)

```
2099 \newdimen\jsc@tocl@width
2100 \newcommand{\tableofcontents}{%
2101 <*book | report>
2102 \settowidth\jsc@tocl@width{\headfont\prechaptername\postchaptername}%
2103 \settowidth\@tempdima{\headfont\appendixname}%
2104 \ifdim\jsc@tocl@width<\@tempdima \setlength\jsc@tocl@width{\@tempdima}\fi
2105 \ifdim\jsc@tocl@width<2\zw \divide\jsc@tocl@width by 2 \advance\jsc@tocl@width 1\zw\fi
2106 \if@twocolumn
2107 \@restonecoltrue\onecolumn
2108 \else
2109 \@restonecolfalse
2110 \fi
2111 \chapter*{\contentsname}%
2112 \@mkboth{\contentsname}{}%
2113 </book | report>
```

```

2114 <{*!book&!report}
2115 \settowidth\jsc@tocl@width{\headfont\presectionname\postsectionname}%
2116 \settowidth\@tempdima{\headfont\appendixname}%
2117 \ifdim\jsc@tocl@width<\@tempdima\relax\setlength\jsc@tocl@width{\@tempdima}\fi
2118 \ifdim\jsc@tocl@width<2\zw \divide\jsc@tocl@width by 2 \advance\jsc@tocl@width 1\zw\fi
2119 \section*{\contentsname}%
2120 \mkboth{\contentsname}{\contentsname}%
2121 </!book&!report}
2122 \@starttoc{toc}%
2123 <book | report> \if@restonecol\twocolumn\fi
2124 }

```

\l@part 部の目次です。

```

2125 \newcommand*{\l@part}[2]{%
2126 \ifnum \c@tocdepth >-2\relax
2127 <|book&!report> \addpenalty\@secpenalty
2128 <book | report> \addpenalty{-\@highpenalty}%
2129 \addvspace{2.25em \@plus\jsc@empt}%
2130 \begingroup
2131 \parindent \z@
2132 % \@pnumwidth should be \@tocrmarg
2133 % \rightskip \@pnumwidth
2134 \rightskip \@tocrmarg
2135 \parfillskip -\rightskip
2136 {\leavevmode
2137 \large \headfont
2138 \setlength\@lnumwidth{4\zw}%
2139 #1\hfil \hb@xt@\@pnumwidth{\hss #2}}\par
2140 \nobreak
2141 <book | report> \global\@nobreaktrue
2142 <book | report> \everypar{\global\@nobreakfalse\everypar{}}%
2143 \endgroup
2144 \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)

```

2145 <{*book | report}
2146 \newcommand*{\l@chapter}[2]{%
2147 \ifnum \c@tocdepth >\m@ne
2148 \addpenalty{-\@highpenalty}%
2149 \addvspace{1.0em \@plus\jsc@empt}
2150 % \vskip 1.0em \@plus\p@ % book.cls では↑がこうなっている
2151 \begingroup
2152 \parindent\z@
2153 % \rightskip\@pnumwidth
2154 \rightskip\@tocrmarg
2155 \parfillskip-\rightskip
2156 \leavevmode\headfont

```

```

2157 % \if@english\setlength\@lnumwidth{5.5em}\else\setlength\@lnumwidth{4.683\zw}\fi
2158 \setlength\@lnumwidth{\jsc@tocl@width}\advance\@lnumwidth 2.683\zw
2159 \advance\leftskip\@lnumwidth \hskip-\leftskip
2160 #1\nobreak\hfil\nobreak\hbox to\@pnumwidth{\hss#2}\par
2161 \penalty\@highpenalty
2162 \endgroup
2163 \fi}
2164 </book | report>

```

\l@section 節の目次です。

```

2165 <*\book&!report>
2166 \newcommand*{\l@section}[2]{%
2167 \ifnum \c@tocdepth >\z@
2168 \addpenalty{\@secpenalty}%
2169 \addvspace{1.0em \@plus\jsc@mpt}%
2170 \begingroup
2171 \parindent\z@
2172 % \rightskip\@pnumwidth
2173 \rightskip\@tocrmarg
2174 \parfillskip-\rightskip
2175 \leavevmode\headfont
2176 %\setlength\@lnumwidth{4\zw}% 元 1.5em [2003-03-02]
2177 \setlength\@lnumwidth{\jsc@tocl@width}\advance\@lnumwidth 2\zw
2178 \advance\leftskip\@lnumwidth \hskip-\leftskip
2179 #1\nobreak\hfil\nobreak\hbox to\@pnumwidth{\hss#2}\par
2180 \endgroup
2181 \fi}
2182 </!\book&!report>

```

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

```

2183 <book | report> % \newcommand*{\l@section}{\@dottedtocline{1}{1\zw}{3.683\zw}}

```

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

```

\l@subparagraph 2184 <*\book&!report>
2185 % \newcommand*{\l@subsection} {\@dottedtocline{2}{1.5em}{2.3em}}
2186 \newcommand*{\l@subsubsection}{\@dottedtocline{3}{3.8em}{3.2em}}
2187 \newcommand*{\l@paragraph} {\@dottedtocline{4}{7.0em}{4.1em}}
2188 % \newcommand*{\l@subparagraph} {\@dottedtocline{5}{10em}{5em}}
2189 %
2190 % \newcommand*{\l@subsection} {\@dottedtocline{2}{1\zw}{3\zw}}
2191 % \newcommand*{\l@subsubsection}{\@dottedtocline{3}{2\zw}{3\zw}}
2192 % \newcommand*{\l@paragraph} {\@dottedtocline{4}{3\zw}{3\zw}}
2193 % \newcommand*{\l@subparagraph} {\@dottedtocline{5}{4\zw}{3\zw}}
2194 %
2195 \newcommand*{\l@subsection}{%

```

```

2196 \@tempdima\jsc@tocl@width \advance\@tempdima -1\zw
2197 \@dottedtocline{2}{\@tempdima}{3\zw}}
2198 \newcommand*{\l@subsubsection}{%
2199 \@tempdima\jsc@tocl@width \advance\@tempdima 0\zw
2200 \@dottedtocline{3}{\@tempdima}{4\zw}}
2201 \newcommand*{\l@paragraph}{%
2202 \@tempdima\jsc@tocl@width \advance\@tempdima 1\zw
2203 \@dottedtocline{4}{\@tempdima}{5\zw}}
2204 \newcommand*{\l@subparagraph}{%
2205 \@tempdima\jsc@tocl@width \advance\@tempdima 2\zw
2206 \@dottedtocline{5}{\@tempdima}{6\zw}}
2207 </!book&!report>
2208 <*book | report>
2209 % \newcommand*{\l@section} {\@dottedtocline{2}{3.8em}{3.2em}}
2210 % \newcommand*{\l@subsubsection}{\@dottedtocline{3}{7.0em}{4.1em}}
2211 % \newcommand*{\l@paragraph} {\@dottedtocline{4}{10em}{5em}}
2212 % \newcommand*{\l@subparagraph} {\@dottedtocline{5}{12em}{6em}}
2213 \newcommand*{\l@section}{%
2214 \@tempdima\jsc@tocl@width \advance\@tempdima -1\zw
2215 \@dottedtocline{1}{\@tempdima}{3.683\zw}}
2216 \newcommand*{\l@section}{%
2217 \@tempdima\jsc@tocl@width \advance\@tempdima 2.683\zw
2218 \@dottedtocline{2}{\@tempdima}{3.5\zw}}
2219 \newcommand*{\l@subsubsection}{%
2220 \@tempdima\jsc@tocl@width \advance\@tempdima 6.183\zw
2221 \@dottedtocline{3}{\@tempdima}{4.5\zw}}
2222 \newcommand*{\l@paragraph}{%
2223 \@tempdima\jsc@tocl@width \advance\@tempdima 10.683\zw
2224 \@dottedtocline{4}{\@tempdima}{5.5\zw}}
2225 \newcommand*{\l@subparagraph}{%
2226 \@tempdima\jsc@tocl@width \advance\@tempdima 16.183\zw
2227 \@dottedtocline{5}{\@tempdima}{6.5\zw}}
2228 </book | report>

```

`\numberline` 欧文版 L^AT_EX では `\numberline{...}` は幅 `\@tempdima` の箱に左詰めで出力する命令です
`\@lnumwidth` 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。

```

2229 \newdimen\@lnumwidth
2230 \def\numberline#1{\hb@xt{\@lnumwidth}{#1\hfil}\hspace{0pt}}

```

`\@dottedtocline` L^AT_EX 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 ·\hss}\hfill}
```


とします。

```
2231 \def\jsTocLine{\leaders\hbox{%
2232   $\m@th \mkern \@dotsep mu\hbox{.}\mkern \@dotsep mu$\hfill}
2233 \def\@dottedtocline#1#2#3#4#5{\ifnum #1>\c@tocdepth \else
2234   \vskip \z@ \@plus.2\jsc@mpt
2235   {\leftskip #2\relax \rightskip \@tocrmarg \parfillskip -\rightskip
2236    \parindent #2\relax\@afterindenttrue
2237    \interlinepenalty\@M
2238    \leavevmode
2239    \@lnumwidth #3\relax
2240    \advance\leftskip \@lnumwidth \null\nobreak\hskip -\leftskip
2241    {#4}\nobreak
2242    \jsTocLine \nobreak\hb@xt@{\pnumwidth}{%
2243      \hfil\normalfont \normalcolor #5}\par}\fi}
```

■ 図目次と表目次

`\listoffigures` 図目次を出力します。

```
2244 \newcommand{\listoffigures}{%
2245   <*book | report>
2246   \if@twocolumn\@restonecoltrue\onecolumn
2247   \else\@restonecolfalse\fi
2248   \chapter*{\listfigurename}%
2249   \@mkboth{\listfigurename}{}%
2250   </book | report>
2251   <!*book&!report>
2252   \section*{\listfigurename}%
2253   \@mkboth{\listfigurename}{\listfigurename}%
2254   <!/book&!report>
2255   \@starttoc{lof}%
2256   <book | report> \if@restonecol\twocolumn\fi
2257 }
```

`\l@figure` 図目次の項目を出力します。

```
2258 \newcommand*{\l@figure}{\@dottedtocline{1}{1\zw}{3.683\zw}}
```

`\listoftables` 表目次を出力します。

```
2259 \newcommand{\listoftables}{%
2260   <*book | report>
2261   \if@twocolumn\@restonecoltrue\onecolumn
2262   \else\@restonecolfalse\fi
2263   \chapter*{\listtablename}%
2264   \@mkboth{\listtablename}{}%
2265   </book | report>
2266   <!*book&!report>
2267   \section*{\listtablename}%
2268   \@mkboth{\listtablename}{\listtablename}%
2269   <!/book&!report>
```

```

2270 \starttoc{lot}%
2271 <book | report> \if@restonecol\twocolumn\fi
2272 }

```

`\l@table` 表目次は図目次と同じです。

```

2273 \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。

```

2274 \newdimen\bibindent
2275 \setlength\bibindent{2\zw}

```

`thebibliography` (*env.*) 参考文献リストを出力します。

```

2276 \newenvironment{thebibliography}[1]{%
2277 \global\let\presectionname\relax
2278 \global\let\postsectionname\relax
2279 <article | jsf> \section*{\refname}\@mkboth{\refname}{\refname}%
2280 <*kiyou>
2281 \vspace{1.5\baselineskip}
2282 \subsubsection*{\refname}\@mkboth{\refname}{\refname}%
2283 \vspace{0.5\baselineskip}
2284 </kiyou>
2285 <book | report> \chapter*{\bibname}\@mkboth{\bibname}{}%
2286 <book | report> \addcontentsline{toc}{chapter}{\bibname}%
2287 \list{\@biblabel{\@arabic\c@enumiv}}%
2288 {\settowidth\labelwidth{\@biblabel{#1}}%
2289 \leftmargin\labelwidth
2290 \advance\leftmargin\labelsep
2291 \@openbib@code
2292 \usecounter{enumiv}%
2293 \let\p@enumiv\@empty
2294 \renewcommand\theenumiv{\@arabic\c@enumiv}}%
2295 <kiyou> \small
2296 \sloppy
2297 \clubpenalty4000
2298 \@clubpenalty\clubpenalty
2299 \widowpenalty4000%
2300 \sfcode`.\@m}
2301 {\def\@noitemerr
2302 {\@latex@warning{Empty `thebibliography' environment}}%
2303 \endlist}

```

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

```

2304 \newcommand{\newblock}{\hspace{.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。

```
2305 \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。

```
2306 % \def\@biblabel#1{\inhibitglue [#1] \inhibitglue}
```

`\cite` 文献の番号を出力する部分は ltbib1.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` で取っているので、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。

```
2307 % \def\@citex[#1]#2{\leavevmode
2308 %   \let\@citea\@empty
2309 %   \@cite{\@for\@citeb:=#2\do
2310 %     {\@citea\def\@citea{\inhibitglue\penalty\@m\ }%
2311 %     \edef\@citeb{\expandafter\@firstofone\@citeb\@empty}%
2312 %     \if@filesw\immediate\write\@auxout{\string\citation{\@citeb}}\fi
2313 %     \ifundefined{b@\@citeb}\mbox{\normalfont\bfseries ?}%
2314 %     \G@refundefinedtrue
2315 %     \latex@warning
2316 %       {Citation '\@citeb' on page \thepage \space undefined}}%
2317 %     {\@citeofmt{\csname b@\@citeb\endcsname}}}\@citeb}}{#1}}
2318 % \def\@cite#1#2{\inhibitglue [{#1\if@tempswa , #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。

```
2319 % \DeclareRobustCommand\cite{\unskip
2320 %   \@ifnextchar [{\@tempswatrue\@citex}{\@tempswafalse\@citex[]}]
2321 %   \def\@cite#1#2{$\sim\hbox{\scriptsize{#1\if@tempswa
2322 %     , \inhibitglue\ #2\fi}} ) }$}
```

10.3 索引

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

```
2323 \newenvironment{theindex}{% 索引を 3 段組で出力する環境
2324   \if@twocolumn
2325     \onecolumn\@restonecolfalse
2326   \else
2327     \clearpage\@restonecoltrue
2328   \fi
2329   \columnseprule.4pt \columnsep 2\zw
2330   \ifx\multicols\@undefined
2331     <book | report> \twocolumn[\@makeschapterhead{\indexname}]%
2332     <book | report> \addcontentsline{toc}{chapter}{\indexname}%
2333     <!book&!report> \def\presectionname{\def\postsectionname}%
2334     <!book&!report> \twocolumn[section*{\indexname}]]%
```

```

2335 \else
2336 \ifdim\textwidth<\fullwidth
2337 \setlength{\evensidemargin}{\oddsidemargin}
2338 \setlength{\textwidth}{\fullwidth}
2339 \setlength{\linewidth}{\fullwidth}
2340 <book | report> \begin{multicols}{3}[chapter*{\indexname}]%
2341 <book | report> \addcontentsline{toc}{chapter}{\indexname}]%
2342 <!book&!report> \def\presectionname{}\def\postsectionname{}%
2343 <!book&!report> \begin{multicols}{3}[section*{\indexname}]%
2344 \else
2345 <book | report> \begin{multicols}{2}[chapter*{\indexname}]%
2346 <book | report> \addcontentsline{toc}{chapter}{\indexname}]%
2347 <!book&!report> \def\presectionname{}\def\postsectionname{}%
2348 <!book&!report> \begin{multicols}{2}[section*{\indexname}]%
2349 \fi
2350 \fi
2351 <book | report> \@mkboth{\indexname}{}%
2352 <!book&!report> \@mkboth{\indexname}{\indexname}%
2353 \plainifnotempty % \thispagestyle{plain}
2354 \parindent\z@
2355 \parskip\z@ \@plus .3\jcs@empt\relax
2356 \let\item\@idxitem
2357 \raggedright
2358 \footnotesize\narrowbaselines
2359 }{
2360 \ifx\multicols\@undefined
2361 \if@restonecol\onecolumn\fi
2362 \else
2363 \end{multicols}
2364 \fi
2365 \clearpage
2366 }

```

`\@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` 2367 `\newcommand{\@idxitem}{\par\hangindent 4\zw} % 元 40pt`
`\subsubitem` 2368 `\newcommand{\subitem}{\@idxitem \hspace*{2\zw}} % 元 20pt`
2369 `\newcommand{\subsubitem}{\@idxitem \hspace*{3\zw}} % 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。

```
2370 \newcommand{\indexspace}{\par \vskip 10\jcs@empt \@plus5\jcs@empt \@minus3\jcs@empt\relax}
```

`\seename` 索引の `\see`, `\seealso` コマンドで出力されるものです。デフォルトはそれぞれ *see*, *see also*

`\alsoname` 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。

```

2371 \newcommand\seename{\if@english see\else →\fi}
2372 \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^AT_EX 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。

[X_YL^AT_EX-j_a]

```
2373 % \@ifl@t@r\pfmtversion{2016/09/03}
2374 %      {\jsc@needspace@tchfalse}{\jsc@needspace@tchtrue}
2375 % \ifjsc@needspace@tch
2376 %   \let\footnotes@ve=\footnote
2377 %   \def\footnote{\inhibitglue\footnotes@ve}
2378 %   \let\footnotemarks@ve=\footnotemark
2379 %   \def\footnotemark{\inhibitglue\footnotemarks@ve}
2380 % \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 `\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_EX 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^AT_EX の変更に追随しました (Thanks: 角藤さん)。pL^AT_EX 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。

[X_YL^AT_EX-j_a]

```
2381 % \@ifl@t@r\pfmtversion{2016/04/17}
2382 %      {\jsc@needspace@tchfalse}{\jsc@needspace@tchtrue}
2383 % \ifjsc@needspace@tch
2384 % \renewcommand\@makefnmark{%
2385 %   \ifdir \hbox{\hbox{\@textsuperscript{\normalfont\@thefnmark}}\hbox{}}%
2386 %   \else\hbox{\yoko\@textsuperscript{\normalfont\@thefnmark}}\fi}
2387 % \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 `newtxtext` や `newpxtext` の使用時におかしくなってしまう。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)。

```
2388 \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。

```
2389 % \def\thefootnote{\ifnum\c@footnote>\z@ 注\kern0.1\zw\@arabic\c@footnote\fi}
```

`\footnoterule` 本文と脚注の間の罫線です。

```
2390 \renewcommand{\footnoterule}{%
2391   \kern-3\jsc@empt
2392   \hrule width .4\columnwidth height 0.4\jsc@empt
2393   \kern 2.6\jsc@empt}
```

`\c@footnote` 脚注番号は章ごとにリセットされます。

```
2394 <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_EX and TUG NEWS*, Vol. 2, No. 4 (1993), p. 9)

[2016-08-25] コミュニティ版 p_ΛT_EX 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_ΛT_EX のバグ修正に追随しました。

[2016-11-29] 古い p_ΛT_EX 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。

```
2395 \long\def\@footnotetext{%
2396   \insert\footins\bgroup
2397     \normalfont\footnotesize
2398     \interlinepenalty\interfootnotelinepenalty
2399     \splittopskip\footnotesep
2400     \splitmaxdepth \dp\strutbox \floatingpenalty \@MM
2401     \hsize\columnwidth \@parboxrestore
2402     \protected@edef\@currentlabel{%
2403       \csname p@footnote\endcsname\@thefnmark
2404     }%
2405     \color@begingroup
2406     \@makefntext{%
2407       \rule\z@\footnotesep\ignorespaces}%
2408     \futurelet\jsc@next\jsc@fo@t}
2409 \def\jsc@fo@t{\ifcat\bgroup\noexpand\jsc@next \let\jsc@next\jsc@fo@t
2410               \else \let\jsc@next\jsc@fo@t\fi \jsc@next}
2411 \def\jsc@fo@t{\bgroup\aftergroup\jsc@@foot\let\jsc@next}
2412 \def\jsc@fo@t#1{#1\jsc@@foot}
2413 \def\jsc@@foot{\@finalstrut\strutbox\color@endgroup\egroup
2414   \ifx\pltx@foot@penalty\@undefined\else
2415     \ifhmode\null\fi
2416     \ifnum\pltx@foot@penalty=\z@\else
2417       \penalty\pltx@foot@penalty
2418       \pltx@foot@penalty\z@
2419     \fi
2420   \fi}
```

`\@makefn`text 実際に脚注を出力する命令です。`\@makefn`mark 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。

```
2421 \newcommand\@makefn
```

text[1]{%
2422 \advance\leftskip 3\zw
2423 \parindent 1\zw
2424 \noindent
2425 \llap{\@makefnmark\hskip0.3\zw}#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。

```
2426 % \def\@xfootnotenext[#1]{%  
2427 %   \begingroup  
2428 %     \ifnum#1>\z@  
2429 %       \csname c@\mpfn\endcsname #1\relax  
2430 %       \unrestored@protected@xdef\@thefnmark{\thempfn}%  
2431 %     \else  
2432 %       \unrestored@protected@xdef\@thefnmark{}%  
2433 %     \fi  
2434 %   \endgroup  
2435 %   \@footnotetext}
```

11 段落の頭へのグルー挿入禁止

段落頭のかぎっこなどを見かけ 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)。

[X_YLA_TE_X-ja]

```
2436 % \def\@inhibitglue{%  
2437 %   \futurelet\@let@token\@@inhibitglue}  
2438 % \begingroup  
2439 % \let\GDEF=\gdef  
2440 % \let\CATCODE=\catcode  
2441 % \let\ENDGROUP=\endgroup  
2442 % \CATCODE`k=12  
2443 % \CATCODE`a=12  
2444 % \CATCODE`n=12
```

```

2445 % \CATCODE`j=12
2446 % \CATCODE`i=12
2447 % \CATCODE`c=12
2448 % \CATCODE`h=12
2449 % \CATCODE`r=12
2450 % \CATCODE`t=12
2451 % \CATCODE`e=12
2452 % \GDEF\KANJI@CHARACTER{kanji character }
2453 % \ENDGROUP
2454 % \def\@@inhibitglue{%
2455 %   \expandafter\expandafter\expandafter\jsc@inhibitglue\expandafter\meaning\expandafter\@le
2456 % \expandafter\def\expandafter\jsc@inhibitglue\expandafter#\expandafter1\KANJI@CHARACTER#2#3
2457 %   \def\jsc@ig@temp{#1}%
2458 %   \ifx\jsc@ig@temp\@empty
2459 %     \ifnum\the\inhibitxspcode`#2=2\relax
2460 %       \inhibitglue
2461 %     \fi
2462 %   \fi}
2463 % \let\everyparhook=\@inhibitglue
2464 % \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。

```

2465 % \def\@doendpe{%
2466 %   \@endpetrue
2467 %   \def\par{%
2468 %     \@restorepar\clubpenalty\clubpenalty\everypar{\everyparhook}\par\@endpefalse}%
2469 %   \everypar{\setbox\z@\lastbox}\everypar{\everyparhook}\@endpefalse\everyparhook}}

```

[2017-08-31] `minipage` 環境にも対策します。

```

2470 % \def\@setminipage{%
2471 %   \@minipagetrue
2472 %   \everypar{\@minipagefalse\everypar{\everyparhook}}%
2473 % }

```

`\item` 命令の直後です。

```

2474 % \def\@item[#1]{%
2475 %   \if@noparitem
2476 %     \@donoparitem
2477 %   \else
2478 %     \if@inlabel
2479 %       \indent \par
2480 %     \fi
2481 %     \ifhmode
2482 %       \unskip\unskip \par
2483 %     \fi

```



```

2484 % \if@newlist
2485 % \if@nobreak
2486 % \@nbitem
2487 % \else
2488 % \addpenalty\@beginparpenalty
2489 % \addvspace\@topsep
2490 % \addvspace{-\parskip}%
2491 % \fi
2492 % \else
2493 % \addpenalty\@itempenalty
2494 % \addvspace\itemsep
2495 % \fi
2496 % \global\@inlabeltrue
2497 % \fi
2498 % \everypar{%
2499 % \@minipagefalse
2500 % \global\@newlistfalse
2501 % \if@inlabel
2502 % \global\@inlabelfalse
2503 % {\setbox\z@\lastbox
2504 % \ifvoid\z@
2505 % \kern-\itemindent
2506 % \fi}%
2507 % \box\@labels
2508 % \penalty\z@
2509 % \fi
2510 % \if@nobreak
2511 % \@nobreakfalse
2512 % \clubpenalty \@M
2513 % \else
2514 % \clubpenalty \@clubpenalty
2515 % \everypar{\everyparhook}%
2516 % \fi\everyparhook}%
2517 % \if@noitemarg
2518 % \@noitemargfalse
2519 % \if@nmbrlist
2520 % \refstepcounter\@listctr
2521 % \fi
2522 % \fi
2523 % \sbox\@tempboxa{\makelabel{#1}}%
2524 % \global\setbox\@labels\hbox{%
2525 % \unhbox\@labels
2526 % \hskip \itemindent
2527 % \hskip -\labelwidth
2528 % \hskip -\labelsep
2529 % \ifdim \wd\@tempboxa >\labelwidth
2530 % \box\@tempboxa
2531 % \else
2532 % \hbox to\labelwidth {\unhbox\@tempboxa}%

```

```

2533 % \fi
2534 % \hskip \labelsep}%
2535 % \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。

```

2536 % \def\@afterheading{%
2537 % \@nobreaktrue
2538 % \everypar{%
2539 % \if@nobreak
2540 % \@nobreakfalse
2541 % \clubpenalty \@M
2542 % \if@afterindent \else
2543 % {\setbox\z@\lastbox}%
2544 % \fi
2545 % \else
2546 % \clubpenalty \@clubpenalty
2547 % \everypar{\everyparhook}%
2548 % \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_{La}T_EX 2_ε 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。

```

2549 % \def\@gnewline #1{%
2550 % \ifvmode
2551 % \@nolnerr
2552 % \else
2553 % \unskip \reserved@e {\reserved@f#1}\nobreak \hfil \break \null
2554 % \inhibitglue \ignorespaces
2555 % \fi}

```

12 いろいろなロゴ

L_AT_EX 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。

```

2556 \if@jslogo
2557 \IfFileExists{jslogo.sty}{%
2558 \RequirePackage{jslogo}%
2559 \def\小{\jslg@small}%
2560 \def\上小{\jslg@uppersmall}%

```

```

2561 }{%
2562   \ClassWarningNoLine{\jsc@clsname}{%
2563     The redefinitions of LaTeX-related logos has\MessageBreak
2564     been moved to jslogo.sty since 2016, but\MessageBreak
2565     jslogo.sty not found. Current release of\MessageBreak
2566     'jsclasses' includes it, so please check\MessageBreak
2567     the installation}%
2568 }
2569 \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 `LATEX` の `\ProvidesFile` で悪さをする例が `FTeX` 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 `ltxclass.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。

```

2570 %\let\ltx@ifnextchar\@ifnextchar
2571 %\def\ProvidesFile#1{%
2572 %  \begingroup
2573 %    \catcode`\ 10 %
2574 %    \ifnum \endlinechar<256 %
2575 %      \ifnum \endlinechar>\m@ne
2576 %        \catcode\endlinechar 10 %
2577 %      \fi
2578 %    \fi
2579 %    \@makeother\/%
2580 %    \@makeother\&%
2581 %    \ltx@ifnextchar[{\@providesfile{#1}}{\@providesfile{#1}[]}]

```

14 初期設定

■いろいろな語

```

\prepartname
\postpartname 2582 \newcommand{\prepartname}{\if@english Part~\else 第\fi}
\prechaptername 2583 \newcommand{\postpartname}{\if@english\else 部\fi}
2584 \newcommand{\prechaptername}{\if@english Chapter~\else 第\fi}
\postchaptername 2585 \newcommand{\postchaptername}{\if@english\else 章\fi}
\presectionname 2586 \newcommand{\presectionname}{}% 第
\postsectionname 2587 \newcommand{\postsectionname}{}% 節

```

```

\contentsname

\listfigurename 2588 \newcommand{\contentsname}{\if@english Contents\else 目次\fi}
\listtablename 2589 \newcommand{\listfigurename}{\if@english List of Figures\else 図目次\fi}
2590 \newcommand{\listtablename}{\if@english List of Tables\else 表目次\fi}

\refname

\bibname 2591 \newcommand{\refname}{\if@english References\else 参考文献\fi}
\indexname 2592 \newcommand{\bibname}{\if@english Bibliography\else 参考文献\fi}
2593 \newcommand{\indexname}{\if@english Index\else 索引\fi}

\figurename

\tablename 2594 \!\jpf\newcommand{\figurename}{\if@english Fig.~\else 図\fi}
2595 \jpf\newcommand{\figurename}{Fig.~}
2596 \!\jpf\newcommand{\tablename}{\if@english Table~\else 表\fi}
2597 \jpf\newcommand{\tablename}{Table~}

\appendixname

\abstractname 2598 % \newcommand{\appendixname}{\if@english Appendix~\else 付録\fi}
2599 \newcommand{\appendixname}{\if@english \else 付録\fi}
2600 \!book\newcommand{\abstractname}{\if@english Abstract\else 概要\fi}

```

■今日の日付 L^AT_EX 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 2022 年 11 月 6 日で、和暦では令和 4 年 11 月 6 日です。

```

\today

2601 \newif\if 西暦 \西暦 true
2602 \def\西暦{\西暦 true}
2603 \def\和暦{\西暦 false}
2604 \newcount\heisei \heisei\year \advance\heisei-1988\relax
2605 \def\pltx@today@year@#1{%
2606   \ifnum\numexpr\year-#1=1 元\else
2607     \ifnum1=\IfDirectionTateTF{1}{0}%
2608       \tokansuji{\year-#1}%
2609     \else
2610       \number\numexpr\year-#1\relax\nobreak
2611     \fi
2612   \fi 年
2613 }
2614 \def\pltx@today@year{%
2615   \ifnum\numexpr\year*10000+\month*100+\day<19890108
2616     昭和\pltx@today@year@{1925}%
2617   \else\ifnum\numexpr\year*10000+\month*100+\day<20190501
2618     平成\pltx@today@year@{1988}%
2619   \else
2620     令和\pltx@today@year@{2018}%
2621   \fi\fi}
2622 \def\today{%

```

```

2623 \if@english
2624 \ifcase\month\or
2625     January\or February\or March\or April\or May\or June\or
2626     July\or August\or September\or October\or November\or December\fi
2627     \space\number\day, \number\year
2628 \else\if 西暦
2629     \ifnum1=\IfDirectionTateTF{1}{0}\tokansuji{\year}
2630     \else\number\year\nobreak\fi 年
2631 \else
2632     \pltx@today@year
2633 \fi
2634 \ifnum1=\IfDirectionTateTF{1}{0}%
2635     \tokansuji{\month}月
2636     \tokansuji{\day}日
2637 \else
2638     \number\month\nobreak 月
2639     \number\day\nobreak 日
2640 \fi\fi}

```

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

```

2641 \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。

```

2642 <article>\if@slide \pagestyle{empty} \else \pagestyle{plain} \fi
2643 <book>\if@report \pagestyle{plain} \else \pagestyle{headings} \fi
2644 <report|kiyou>\pagestyle{plain}
2645 <jspf>\pagestyle{headings}
2646 \pagenumbering{arabic}
2647 \if@twocolumn
2648     \twocolumn
2649     \sloppy
2650     \flushbottom
2651 \else
2652     \onecolumn
2653     \raggedbottom
2654 \fi
2655 \if@slide
2656     \renewcommand\kanjifamilydefault{\gtdefault}
2657     \renewcommand\familydefault{\sfdefault}
2658     \raggedright
2659     \setxkanjiskip{0.1em}
2660 \fi

```

15 実験的コード

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

```
2661 <*article | book | report>
2662 % \ifpackageloaded{exppl2e}{\jsc@needspace@tchtrue}{\jsc@needspace@tchfalse}
2663 % \ifjsc@needspace@tch\else
2664 %   \expandafter\endinput
2665 % \fi
2666 \endinput
```

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
2667 \def\@gnewline #1{%
2668   \ifvmode
2669     \@nolnerr
2670   \else
2671     \unskip \reserved@a {\reserved@f#1}\nobreak \hfil \break \hskip \z@
2672     \ignorespaces
2673   \fi}
2674 </article | book | report>
2675 </class>
```

以上です。