

pL^AT_EX 2_ε 新ドキュメントクラス

奥村晴彦, 日本語 T_EX 開発コミュニティ

2018/12/11

1 はじめに

これは L^AT_EX3 Project の `classes.dtx` と株式会社アスキーの `jclasses.dtx` に基づいてもともと奥村晴彦により作成されたものです。現在は日本語 T_EX 開発コミュニティにより GitHub で管理されています。

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

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

[2016-07-13] 日本語 T_EX 開発コミュニティによる管理に移行しました。

[2009-02-22] 田中琢爾氏による upL^AT_EX 対応パッチを取り込みました。

ここでは次のドキュメントクラス (スタイルファイル) を作ります。

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

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

L^AT_EX 2_ε あるいは pL^AT_EX 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 (T_EX フォントメトリック) として東京書籍印刷の小林肇さんの作られた JIS フォントメトリック `jis.tfm`, `jisg.tfm` を標準で使います。従来のフォントメトリック `min10.tfm`, `goth10.tfm` の類を使うには

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

のように `mingoth` オプションを付けます。

■サイズオプションの扱いが違う 標準のドキュメントクラスでは本文のポイント数を指定するオプションがありましたが、ポイント数は 10, 11, 12 しかなく、それぞれ別のクラスオプションファイルを読み込むようになっていました。しかも、標準の 10 ポイント以外では多少フォントのバランスが崩れることがあり、あまり便利ではありませんでした。ここでは文字サイズを増すとページを小さくし、 \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 \article\def\jsc@clsname{jsarticle}
2 \book\def\jsc@clsname{jsbook}
3 \report\def\jsc@clsname{jsreport}
4 \jspf\def\jsc@clsname{jspf}
5 \kiyou\def\jsc@clsname{kiyou}
```

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

```
6 \newif\ifjsc@needsp@tch
7 \jsc@needsp@tchfalse
```

2 オプション

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

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

`\ifrestonecol` 段組のときに真になる論理変数です。

```
8 \newif\ifrestonecol
```

`\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。

```
9 \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。

```
10 \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。

```
11 \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。

```
12 \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。

```
13 \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 1m^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 1.5m^2 ですが、ISO では B1 判の辺の長さが A0 判と A1 判の辺の長さの幾何平均です。したがって ISO の B0 判は $1000\text{mm} \times 1414\text{mm}$ です。このため、 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$ の `b5paper` は $250\text{mm} \times 176\text{mm}$ ですが、 $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$ の `b5paper` は $257\text{mm} \times 182\text{mm}$ になっています。ここでは $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$ にならって JIS に従いました。

デフォルトは `a4paper` です。

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

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

```

44 \DeclareOption{b5j}{%
45   \setlength\paperheight {257mm}%
46   \setlength\paperwidth  {182mm}}
47 \DeclareOption{a4var}{%
48   \setlength\paperheight {283mm}%
49   \setlength\paperwidth  {210mm}}
50 \DeclareOption{b5var}{%
51   \setlength\paperheight {230mm}%
52   \setlength\paperwidth  {182mm}}
53 \DeclareOption{letterpaper}{%
54   \setlength\paperheight {11in}%
55   \setlength\paperwidth  {8.5in}}
56 \DeclareOption{legalpaper}{%
57   \setlength\paperheight {14in}%
58   \setlength\paperwidth  {8.5in}}
59 \DeclareOption{executivepaper}{%
60   \setlength\paperheight {10.5in}%
61   \setlength\paperwidth  {7.25in}}

```

■横置き 用紙の縦と横の長さを入れ換えます。

```

62 \newif\if@landscape
63 \@landscapefalse
64 \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もので、簡単のため article のみで使えるオプションとしました。

```

65 \newif\if@slide
66 \@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。

```

67 \newcommand{\@ptsize}{0}
68 \newif\ifjpsc@mag\jsc@magtrue
69 \newif\ifjpsc@mag@xreal\jsc@mag@xrealfalse
70 \def\jsc@magscale{1}

```

```

71 (*article)
72 \DeclareOption{slide}{%
73   \@slidetrue\def\jsc@magscale{3.583}
74   \renewcommand{\@ptsize}{26}
75   \@landscapetrue\@titlepagetrue}
76 (/article)
77 \DeclareOption{8pt}{\def\jsc@magscale{0.833}\renewcommand{\@ptsize}{-2}}
78 \DeclareOption{9pt}{\def\jsc@magscale{0.913}\renewcommand{\@ptsize}{-1}}
79 \DeclareOption{10pt}{\def\jsc@magscale{1}\renewcommand{\@ptsize}{0}}
80 \DeclareOption{11pt}{\def\jsc@magscale{1.095}\renewcommand{\@ptsize}{1}}
81 \DeclareOption{12pt}{\def\jsc@magscale{1.200}\renewcommand{\@ptsize}{2}}
82 \DeclareOption{14pt}{\def\jsc@magscale{1.440}\renewcommand{\@ptsize}{4}}
83 \DeclareOption{17pt}{\def\jsc@magscale{1.728}\renewcommand{\@ptsize}{7}}
84 \DeclareOption{20pt}{\def\jsc@magscale{2}\renewcommand{\@ptsize}{10}}
85 \DeclareOption{21pt}{\def\jsc@magscale{2.074}\renewcommand{\@ptsize}{11}}
86 \DeclareOption{25pt}{\def\jsc@magscale{2.488}\renewcommand{\@ptsize}{15}}
87 \DeclareOption{30pt}{\def\jsc@magscale{2.986}\renewcommand{\@ptsize}{20}}
88 \DeclareOption{36pt}{\def\jsc@magscale{3.583}\renewcommand{\@ptsize}{26}}
89 \DeclareOption{43pt}{\def\jsc@magscale{4.300}\renewcommand{\@ptsize}{33}}
90 \DeclareOption{12Q}{\def\jsc@magscale{0.923}\renewcommand{\@ptsize}{1200}}
91 \DeclareOption{14Q}{\def\jsc@magscale{1.077}\renewcommand{\@ptsize}{1400}}
92 \DeclareOption{10ptj}{\def\jsc@magscale{1.085}\renewcommand{\@ptsize}{1001}}
93 \DeclareOption{10.5ptj}{\def\jsc@magscale{1.139}\renewcommand{\@ptsize}{1051}}
94 \DeclareOption{11ptj}{\def\jsc@magscale{1.194}\renewcommand{\@ptsize}{1101}}
95 \DeclareOption{12ptj}{\def\jsc@magscale{1.302}\renewcommand{\@ptsize}{1201}}
96 \DeclareOption{usemag}{\jsc@magtrue\jsc@mag@xrealfalse}
97 \DeclareOption{nomag}{\jsc@magfalse\jsc@mag@xrealfalse}
98 \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。

```

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

```

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

```
113 \DeclareOption{mentuke}{%
114   \tombowtrue \tombowdatefalse
115   \setlength{\@tombowwidth}{\z@}%
116   \maketombowbox}
```

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

```
117 \DeclareOption{oneside}{\@twosidefalse \@mparswitchfalse}
118 \DeclareOption{twoside}{\@twosidetrue \@mparswitchtrue}
119 \DeclareOption{vartwoside}{\@twosidetrue \@mparswitchfalse}
```

■二段組 `twocolumn` で二段組になります。

```
120 \DeclareOption{onecolumn}{\@twocolumnfalse}
121 \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。

```
122 \DeclareOption{titlepage}{\@titlepagetrue}
123 \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。

```
124 (book|report)\DeclareOption{openright}{\@openrighttrue\@openleftfalse}
125 (book|report)\DeclareOption{openleft}{\@openlefttrue\@openrightfalse}
126 (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` \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。

```
127 \def\eqnarray{%
128   \stepcounter{equation}%
129   \def\@currentlabel{\p@equation\theequation}%
130   \global\@eqnswtrue
131   \m@th
132   \global\@eqcnt\z@
133   \tabskip\@centering}
```

```

134 \let\\@eqnocr
135 $$\everycr{}\halign to\displaywidth\bgroup
136 \hskip\@centering$\displaystyle\tabskip\z@skip{##}$\@eqnse1
137 &\global\@eqcnt\@ne \hfil$\displaystyle{##}\hfil
138 &\global\@eqcnt\tw@ $\displaystyle{##}\hfil\tabskip\@centering
139 &\global\@eqcnt\thr@@ \hb@xt@\z@\bgroup\hss##\egroup
140 \tabskip\z@skip
141 \cr}

```

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

```

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

```

■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。

```

169 % \DeclareOption{openbib}{%
170 % \AtEndOfPackage{%
171 % \renewcommand\@openbib@code{%
172 % \advance\leftmargin\bibindent
173 % \itemindent -\bibindent
174 % \listparindent \itemindent
175 % \parsep \z@}%

```

```
176 % \renewcommand\newblock{\par}}
```

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

```
177 \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。

```
178 \DeclareOption{draft}{\setlength\overfullrule{5pt}}
```

```
179 \DeclareOption{final}{\setlength\overfullrule{0pt}}
```

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

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

```
180 \newif\ifmingoth
```

```
181 \mingothfalse
```

```
182 \newif\ifjisfont
```

```
183 \jisfontfalse
```

```
184 \newif\if@jsc@uplatex
```

```
185 \@jsc@uplatexfalse
```

```
186 \newif\if@jsc@autodetect
```

```
187 \@jsc@autodetectfalse
```

```
188 \DeclareOption{winjis}{%
```

```
189 \ClassWarningNoLine{\jsc@clsname}{%
```

```
190 The option ‘winjis’ has been removed;\MessageBreak
```

```
191 Use ‘\string\usepackage{winjis}’ instead}}
```

```
192 \DeclareOption{mingoth}{\mingothtrue}
```

```
193 \DeclareOption{jis}{\jisfonttrue}
```

```
194 \DeclareOption{uplatex}{\@jsc@uplatextrue}
```

```
195 \DeclareOption{autodetect-engine}{\@jsc@autodetecttrue}
```

```
196 \def\jsc@JYn{\if@jsc@uplatex JY2\else JY1\fi}
```

```
197 \def\jsc@JTn{\if@jsc@uplatex JT2\else JT1\fi}
```

```
198 \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。

```
199 \newif\ifpapersize
200 \papersizefalse
201 \DeclareOption{papersize}{\papersizetrue}
```

■英語化 オプション english を新設しました。

```
202 \newif\if@english
203 \@englishfalse
204 \DeclareOption{english}{\@englishttrue}
```

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

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

```
205 (*book)
206 \newif\if@report
207 \@reportfalse
208 \DeclareOption{report}{\@reporttrue\@openrightfalse\@twosidefalse\@mparswitchfalse}
209 </book>
```

■jslogo パッケージの読み込み L^AT_EX 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。

```
210 \newif\if@jslogo \@jslogotrue
211 \DeclareOption{jslogo}{\@jslogotrue}
212 \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。

```
213 (article)\ExecuteOptions{a4paper,oneside,onecolumn,notitlepage,final}
214 (book)\ExecuteOptions{a4paper,twoside,onecolumn,titlepage,openright,final}
215 (report)\ExecuteOptions{a4paper,oneside,onecolumn,titlepage,openany,final}
216 (jspf)\ExecuteOptions{a4paper,twoside,twocolumn,notitlepage,fleqn,final}
217 (kiyou)\ExecuteOptions{a4paper,twoside,twocolumn,notitlepage,final}
218 \ProcessOptions
```

後処理

```
219 \if@slide
220 \def\maybeblue{\@ifundefined{ver@color.sty}{\color{blue}}
221 \fi
222 \if@landscape
223 \setlength\@tempdima {\paperheight}
224 \setlength\paperheight{\paperwidth}
225 \setlength\paperwidth {\@tempdima}
226 \fi
```

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

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

```
227 \ifnum \ifx\ucs\@undefined\z\else\ucs"3000 \fi ="3000
228 \if@jsc@autodetect
229 \ClassInfo\jsc@clsname{Autodetected engine: upLaTeX}
230 \@jsc@uplatextrue
231 \g@addto@macro\@classoptionslist{,uplatex}
232 \fi
233 \if@jsc@uplatex\else
234 \ClassError\jsc@clsname
235 {You are running upLaTeX.\MessageBreak
236 Please use pLaTeX instead, or add 'uplatex' to\MessageBreak
237 the class option list}
238 {\@ehc}
239 \@jsc@uplatextrue
240 \fi
```

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

```
241 \else
242 \if@jsc@uplatex
243 \ClassError\jsc@clsname
244 {You are running pLaTeX.\MessageBreak
245 Please use upLaTeX instead, or remove 'uplatex' from\MessageBreak
246 the class option list}
247 {\@ehc}
248 \@jsc@uplatexfalse
249 \fi
250 \if@jsc@autodetect
251 \ClassInfo\jsc@clsname{Autodetected engine: pLaTeX}
252 \@jsc@uplatexfalse
253 \fi
254 \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

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

というオプションを与えて両側 0.5 インチのトンボにするといいでしょう。

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

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

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

```
255 \iftombow
256   \newdimen\stockwidth \newdimen\stockheight
257   \setlength{\stockwidth}{\paperwidth}
258   \setlength{\stockheight}{\paperheight}
259   \advance \stockwidth 2in
260   \advance \stockheight 2in
261 \fi
262 \ifpapersize
263   \iftombow
264     \AtBeginDvi{\special{papersize=\the\stockwidth,\the\stockheight}}
265   \else
266     \AtBeginDvi{\special{papersize=\the\paperwidth,\the\paperheight}}
267 \fi
268 \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。

```
269 <article | book | report>\if@slide\def\n@baseline{13}\else\def\n@baseline{16}\fi
270 <jspf>\def\n@baseline{14.554375}
271 <kiyou>\def\n@baseline{14.897}
```

■拡大率の設定 サイズの変更は T_EX 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。

```
272 \newdimen\jsc@empt
273 \newdimen\jsc@mmm
```

```

274 \def\inv@mag{1}
275 \ifjsc@mag
276   \jsc@mpt=1\p@
277   \jsc@mmm=1mm
278   \ifnum\@ptsize=-2
279     \mag 833
280     \def\inv@mag{1.20048}
281     \def\n@baseline{15}%
282   \fi
283   \ifnum\@ptsize=-1
284     \mag 913 % formerly 900
285     \def\inv@mag{1.09529}
286     \def\n@baseline{15}%
287   \fi
288   \ifnum\@ptsize=1
289     \mag 1095 % formerly 1100
290     \def\inv@mag{0.913242}
291   \fi
292   \ifnum\@ptsize=2
293     \mag 1200
294     \def\inv@mag{0.833333}
295   \fi
296   \ifnum\@ptsize=4
297     \mag 1440
298     \def\inv@mag{0.694444}
299   \fi
300   \ifnum\@ptsize=7
301     \mag 1728
302     \def\inv@mag{0.578704}
303   \fi
304   \ifnum\@ptsize=10
305     \mag 2000
306     \def\inv@mag{0.5}
307   \fi
308   \ifnum\@ptsize=11
309     \mag 2074
310     \def\inv@mag{0.48216}
311   \fi
312   \ifnum\@ptsize=15
313     \mag 2488
314     \def\inv@mag{0.401929}
315   \fi
316   \ifnum\@ptsize=20
317     \mag 2986
318     \def\inv@mag{0.334896}
319   \fi
320   \ifnum\@ptsize=26
321     \mag 3583
322     \def\inv@mag{0.279096}

```

```

323 \fi
324 \ifnum\@ptsize=33
325 \mag 4300
326 \def\inv@mag{0.232558}
327 \fi
328 \ifnum\@ptsize=1200
329 \mag 923
330 \def\inv@mag{1.0834236}
331 \fi
332 \ifnum\@ptsize=1400
333 \mag 1077
334 \def\inv@mag{0.928505}
335 \fi
336 \ifnum\@ptsize=1001
337 \mag 1085
338 \def\inv@mag{0.921659}
339 \fi
340 \ifnum\@ptsize=1051
341 \mag 1139
342 \def\inv@mag{0.877963}
343 \fi
344 \ifnum\@ptsize=1101
345 \mag 1194
346 \def\inv@mag{0.837521}
347 \fi
348 \ifnum\@ptsize=1201
349 \mag 1302
350 \def\inv@mag{0.768049}
351 \fi
352 \else
353 \jsc@mpt=\jsc@magscale\p@
354 \jsc@mmm=\jsc@magscale mm
355 \def\inv@mag{1}
356 \ifnum\@ptsize=-2
357 \def\n@baseline{15}%
358 \fi
359 \ifnum\@ptsize=-1
360 \def\n@baseline{15}%
361 \fi
362 \fi
363 (*kiyou)
364 \def\jsc@magscale{0.9769230}
365 \ifjsc@mag
366 \mag 977
367 \def\inv@mag{1.02354}
368 \jsc@mpt=1\p@
369 \jsc@mmm=1mm
370 \else
371 \jsc@mpt=\jsc@magscale\p@

```

```

372 \jsc@mmm=\jsc@magscale mm
373 \def\inv@mag{1}
374 \fi
375 </kiyou>
376 \ifjsc@mag@xreal
377 \RequirePackage{type1cm}
378 \mathchardef\jsc@csta=259
379 \def\jsc@invscale#1#2{%
380 \begingroup \@tempdima=#1\relax \@tempdimb#2\p@ \relax
381 \@tempcnta\@tempdima \multiply\@tempcnta\@ccclvi
382 \divide\@tempcnta\@tempdimb \multiply\@tempcnta\@ccclvi
383 \@tempcntb\p@ \divide\@tempcntb\@tempdimb
384 \advance\@tempcnta-\@tempcntb \advance\@tempcnta-\tw@
385 \@tempdimb\@tempcnta\@ne
386 \advance\@tempcnta\@tempcntb \advance\@tempcnta\@tempcntb
387 \advance\@tempcnta\jsc@csta \@tempdimc\@tempcnta\@ne
388 \@whiledim\@tempdimb<\@tempdimc\do{%
389 \@tempcntb\@tempdimb \advance\@tempcntb\@tempdimc
390 \advance\@tempcntb\@ne \divide\@tempcntb\tw@
391 \ifdim #2\@tempcntb>\@tempdima
392 \advance\@tempcntb\m@ne \@tempdimc=\@tempcntb\@ne
393 \else \@tempdimb=\@tempcntb\@ne \fi}%
394 \xdef\jsc@gtmpa{\the\@tempdimb}%
395 \endgroup #1=\jsc@gtmpa\relax}
396 \expandafter\let\csname OT1/cmr/m/n/10\endcsname\relax
397 \expandafter\let\csname OMX/cmex/m/n/10\endcsname\relax
398 \let\jsc@get@external@font\get@external@font
399 \def\get@external@font{%
400 \jsc@preadjust@extract@font
401 \jsc@get@external@font}
402 \def\jsc@fstrunc#1{%
403 \edef\jsc@tmpa{\strip@pt#1}%
404 \expandafter\jsc@fstrunc@a\jsc@tmpa.****\@nil}
405 \def\jsc@fstrunc@a#1.#2#3#4#5#6\@nil{%
406 \if#5*\else
407 \edef\jsc@tmpa{#1%
408 \ifnum#2#3>\z@ .#2\ifnum#3>\z@ #3\fi\fi}%
409 \fi}
410 \def\jsc@preadjust@extract@font{%
411 \let\jsc@req@size\f@size
412 \dimen@f@size\p@ \jsc@invscale\dimen@\jsc@magscale
413 \advance\dimen@.005pt\relax \jsc@fstrunc\dimen@
414 \let\jsc@ref@size\jsc@tmpa
415 \let\f@size\jsc@ref@size}
416 \def\execute@size@function#1{%
417 \let\jsc@cref@size\f@size
418 \let\f@size\jsc@req@size
419 \csname s@fct@#1\endcsname}
420 \let\jsc@DeclareErrorFont\DeclareErrorFont

```

```

421 \def\DeclareErrorFont#1#2#3#4#5{%
422   \@tempdimc#5\p@ \@tempdimc\jsc@magscale\@tempdimc
423   \edef\jsc@tmpa{{#1}{#2}{#3}{#4}{\strip@pt\@tempdimc}}
424   \expandafter\jsc@DeclareErrorFont\jsc@tmpa}
425 \def\gen@sfcnt{%
426   \edef\mandatory@arg{\mandatory@arg\jsc@cref@size}%
427   \empty@sfcnt}
428 \def\genb@sfcnt{%
429   \edef\mandatory@arg{%
430     \mandatory@arg\expandafter\genb@x\jsc@cref@size..\@@}%
431   \empty@sfcnt}
432 \DeclareErrorFont{OT1}{cmr}{m}{n}{10}
433 \fi

```

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

```

\jsc@smallskip
\jsc@medskip 434 \def\jsc@smallskip{\vspace\jsc@smallskipamount}
\jsc@bigskip 435 %\def\jsc@medskip{\vspace\jsc@medskipamount}
436 %\def\jsc@bigskip{\vspace\jsc@bigskipamount}

```

```

\jsc@smallskipamount
\jsc@medskipamount 437 \newskip\jsc@smallskipamount
\jsc@bigskipamount 438 \jsc@smallskipamount=3\jsc@mpt plus 1\jsc@mpt minus 1\jsc@mpt
439 %\newskip\jsc@medskipamount
440 %\jsc@medskipamount =6\jsc@mpt plus 2\jsc@mpt minus 2\jsc@mpt
441 %\newskip\jsc@bigskipamount
442 %\jsc@bigskipamount =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。

```

443 \setlength\paperwidth{\inv@mag\paperwidth}%
444 \setlength\paperheight{\inv@mag\paperheight}%
445 \iftombow
446   \setlength\stockwidth{\inv@mag\stockwidth}%
447   \setlength\stockheight{\inv@mag\stockheight}%
448 \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。

```

449 % \ifpapersize

```

```

450 % \setlength{\@tempdima}{\paperwidth}
451 % \setlength{\@tempdimb}{\paperheight}
452 % \iftombow
453 %     \advance \@tempdima 2truein
454 %     \advance \@tempdimb 2truein
455 % \fi
456 % \AtBeginDvi{\special{pdf: pagesize width \the\@tempdima\space height \the\@tempdimb}}
457 % \fi

```

3 和文フォントの変更

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

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

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

[2018-02-04] 上記のと通りの「クラスファイルが意図する和文スケール値 (1zw ÷ 要求サイズ)」を表す実数値マクロ `\Cjascale` を定義します。このマクロが定義されている場合, OTF パッケージ (2018/02/01 以降のバージョン) はこれに従います。jsarticle, jsbook, jsreport では, $9.62216 \text{ pt} * 0.961 / 10 \text{ pt} = 0.924690$ です。

```

458 (*!jspf)
459 \def\Cjascale{0.924690}
460 \ifmingoth
461 \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.961] \jsc@pfx@ min10}{}
462 \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.961] \jsc@pfx@ goth10}{}
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 \else
466 \ifjisfont
467 \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.961] \jsc@pfx@ jis}{}
468 \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.961] \jsc@pfx@ jisg}{}
469 \DeclareFontShape{\jsc@JTn}{mc}{m}{n}{<-> s * [0.961] \jsc@pfx@ tmin10}{}
470 \DeclareFontShape{\jsc@JTn}{gt}{m}{n}{<-> s * [0.961] \jsc@pfx@ tgoth10}{}
471 \else
472 \if@jsc@uplatex

```

```

473 \DeclareFontShape{JY2}{mc}{m}{n}{<-> s * [0.924690] upjisr-h}{}
474 \DeclareFontShape{JY2}{gt}{m}{n}{<-> s * [0.924690] upjisg-h}{}
475 \DeclareFontShape{JT2}{mc}{m}{n}{<-> s * [0.924690] upjisr-v}{}
476 \DeclareFontShape{JT2}{gt}{m}{n}{<-> s * [0.924690] upjisg-v}{}
477 \else
478 \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.961] \jsc@pfx@ jis}{}
479 \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.961] \jsc@pfx@ jisg}{}
480 \DeclareFontShape{\jsc@JTn}{mc}{m}{n}{<-> s * [0.961] \jsc@pfx@ tmin10}{}
481 \DeclareFontShape{\jsc@JTn}{gt}{m}{n}{<-> s * [0.961] \jsc@pfx@ tgoth10}{}
482 \fi
483 \fi
484 \fi
485 </!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。

```

486 *jspf)
487 \def\Cjascale{0.903375}
488 \ifmingoth
489 \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ min10}{}
490 \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ goth10}{}
491 \DeclareFontShape{\jsc@JTn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tmin10}{}
492 \DeclareFontShape{\jsc@JTn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tgoth10}{}
493 \else
494 \ifjisfont
495 \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ jis}{}
496 \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ jisg}{}
497 \DeclareFontShape{\jsc@JTn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tmin10}{}
498 \DeclareFontShape{\jsc@JTn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tgoth10}{}
499 \else
500 \if@jsc@uplatex
501 \DeclareFontShape{JY2}{mc}{m}{n}{<-> s * [0.903375] upjisr-h}{}
502 \DeclareFontShape{JY2}{gt}{m}{n}{<-> s * [0.903375] upjisg-h}{}
503 \DeclareFontShape{JT2}{mc}{m}{n}{<-> s * [0.903375] upjisr-v}{}
504 \DeclareFontShape{JT2}{gt}{m}{n}{<-> s * [0.903375] upjisg-v}{}
505 \else
506 \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ jis}{}
507 \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ jisg}{}
508 \DeclareFontShape{\jsc@JTn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tmin10}{}
509 \DeclareFontShape{\jsc@JTn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tgoth10}{}
510 \fi
511 \fi
512 \fi
513 </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ふ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、`TEX` 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よ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。

```

514 % \DeclareFontShape{\jsc@JYn}{mc}{bx}{n}{<->ssub*gt/m/n}{ } % in \jsc@JYnmc
515 % \DeclareFontShape{\jsc@JYn}{gt}{bx}{n}{<->ssub*gt/m/n}{ } % in \jsc@JYngt
516 \DeclareFontShape{\jsc@JYn}{mc}{m}{it}{<->ssub*mc/m/n}{ }
517 \DeclareFontShape{\jsc@JYn}{mc}{m}{sl}{<->ssub*mc/m/n}{ }
518 \DeclareFontShape{\jsc@JYn}{mc}{m}{sc}{<->ssub*mc/m/n}{ }
519 \DeclareFontShape{\jsc@JYn}{gt}{m}{it}{<->ssub*gt/m/n}{ }
520 \DeclareFontShape{\jsc@JYn}{gt}{m}{sl}{<->ssub*gt/m/n}{ }
521 \DeclareFontShape{\jsc@JYn}{mc}{bx}{it}{<->ssub*gt/m/n}{ }
522 \DeclareFontShape{\jsc@JYn}{mc}{bx}{sl}{<->ssub*gt/m/n}{ }
523 % \DeclareFontShape{\jsc@JTn}{mc}{bx}{n}{<->ssub*gt/m/n}{ } % in \jsc@JTnmc
524 % \DeclareFontShape{\jsc@JTn}{gt}{bx}{n}{<->ssub*gt/m/n}{ } % in \jsc@JTngt
525 \DeclareFontShape{\jsc@JTn}{mc}{m}{it}{<->ssub*mc/m/n}{ }
526 \DeclareFontShape{\jsc@JTn}{mc}{m}{sl}{<->ssub*mc/m/n}{ }
527 \DeclareFontShape{\jsc@JTn}{mc}{m}{sc}{<->ssub*mc/m/n}{ }
528 \DeclareFontShape{\jsc@JTn}{gt}{m}{it}{<->ssub*gt/m/n}{ }
529 \DeclareFontShape{\jsc@JTn}{gt}{m}{sl}{<->ssub*gt/m/n}{ }
530 \DeclareFontShape{\jsc@JTn}{mc}{bx}{it}{<->ssub*gt/m/n}{ }
531 \DeclareFontShape{\jsc@JTn}{mc}{bx}{sl}{<->ssub*gt/m/n}{ }
532 \DeclareRobustCommand\rmfamily
533     {\not@math@alphabet\rmfamily\mathrm
534     \romanfamily\rmdefault\kanjifamily\mcdefault\selectfont}
535 \DeclareRobustCommand\sffamily
536     {\not@math@alphabet\sffamily\mathsf
537     \romanfamily\sfddefault\kanjifamily\gtdefault\selectfont}
538 \DeclareRobustCommand\ttfamily
539     {\not@math@alphabet\ttfamily\mathtt
540     \romanfamily\ttdefault\kanjifamily\gtdefault\selectfont}

```

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

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

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

[2017-09-19] 2010 年の `pTEX` の修正で、イタリック補正と和欧文間の `\xkanjiskip` の衝

突が起きなくなっていますから、もうここにあるような単純化は必要ありません。ただし、このクラスファイルが古い TeX 環境で利用される可能性も捨てきれないので、とりあえず残しておきます。

```

541 \ifx\DeclareFixJFMCJKTextFontCommand\undefined
542 \DeclareRobustCommand\textmc[1]{%
543   \relax\ifmmode \expandafter\nfss@text \fi{\mcfamily #1}}
544 \DeclareRobustCommand\textgt[1]{%
545   \relax\ifmmode \expandafter\nfss@text \fi{\gtfamily #1}}
546 \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。

```

547 \def\reDeclareMathAlphabet#1#2#3{%
548   \edef\@tempa{\expandafter@gobble\string#2}%
549   \edef\@tempb{\expandafter@gobble\string#3}%
550   \edef\@tempc{\string @\expandafter@gobbletwo\string#2}%
551   \ifx\@tempc\@tempa%
552     \edef\@tempa{\expandafter@gobbletwo\string#2}%
553     \edef\@tempb{\expandafter@gobbletwo\string#3}%
554   \fi
555   \begingroup
556     \let\protect\noexpand
557     \def\@tempaa{\relax}%
558     \expandafter\ifx\csname RDMAorg@\@tempa\endcsname\relax
559       \edef\@tempaa{\expandafter\def\expandafter\noexpand%
560         \csname RDMAorg@\@tempa\endcsname{%
561           \expandafter\noexpand\csname\@tempa\endcsname}}%
562     \fi
563     \def\@tempbb{\relax}%
564     \expandafter\ifx\csname RDMAorg@\@tempb\endcsname\relax
565       \edef\@tempbb{\expandafter\def\expandafter\noexpand%
566         \csname RDMAorg@\@tempb\endcsname{%
567           \expandafter\noexpand\csname\@tempb\endcsname}}%
568     \fi
569     \edef\@tempc{\@tempaa\@tempbb}%
570     \expandafter\endgroup\@tempc%
571     \edef#1{\noexpand\protect\expandafter\noexpand\csname%
572       \expandafter@gobble\string#1\space\space\endcsname}%
573     \expandafter\edef\csname\expandafter@gobble\string#1\space\space\endcsname%
574     {\noexpand\DualLang@mathalph@bet%
575       {\expandafter\noexpand\csname RDMAorg@\@tempa\endcsname}%
576       {\expandafter\noexpand\csname RDMAorg@\@tempb\endcsname}}%
577   }%

```

```

578 }
579 \@onlypreamble\reDeclareMathAlphabet
580 \def\DualLang@mathalph@bet#1#2{%
581   \relax\ifmmode
582     \ifx\math@bgroup\bgroup%      2e normal style (\mathrm{...})
583       \bgroup\let\DualLang@Mfontsw\DLMfontsw@standard
584     \else
585       \ifx\math@bgroup\relax%      2e two letter style (\rm->\mathrm)
586         \let\DualLang@Mfontsw\DLMfontsw@oldstyle
587       \else
588         \ifx\math@bgroup\@empty%  2.09 oldfont style ({\mathrm ...})
589           \let\DualLang@Mfontsw\DLMfontsw@oldfont
590         \else%                      panic! assume 2e normal style
591           \bgroup\let\DualLang@Mfontsw\DLMfontsw@standard
592         \fi
593       \fi
594     \fi
595   \else
596     \let\DualLang@Mfontsw\@firstoftwo
597   \fi
598   \DualLang@Mfontsw{#1}{#2}%
599 }
600 \def\DLMfontsw@standard#1#2#3{#1{#2{#3}}\egroup}
601 \def\DLMfontsw@oldstyle#1#2{#1\relax\@fontswitch\relax{#2}}
602 \def\DLMfontsw@oldfont#1#2{#1\relax#2\relax}
603 \if@enablejfam
604   \DeclareSymbolFont{mincho}{\jsc@JYn}{mc}{m}{n}
605   \DeclareSymbolFontAlphabet{\mathmc}{mincho}
606   \SetSymbolFont{mincho}{bold}{\jsc@JYn}{gt}{m}{n}
607   \jfam\symmincho
608   \DeclareMathAlphabet{\mathgt}{\jsc@JYn}{gt}{m}{n}
609   \AtBeginDocument{%
610     \reDeclareMathAlphabet{\mathrm}{\@mathrm}{\@mathmc}
611     \reDeclareMathAlphabet{\mathbf}{\@mathbf}{\@mathgt}}
612 \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。

```
613 % \DeclareTextCommand{\textsterling}{OT1}{\itshape\char‘\$}
```

禁則パラメータも若干修正します。

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

設定します。

```
614 \prebreakpenalty\jis"2147=10000 % 5000 '
615 \postbreakpenalty\jis"2148=10000 % 5000 "
616 \prebreakpenalty\jis"2149=10000 % 5000 "
```

「 \TeX !」「 $\bar{5}15$ 」の記号と数字の間に四分アキが入らないようにします。

```
617 \inhibitxspcode'!=1
618 \inhibitxspcode'\bar{5}=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

```
619 % \xspcode'.=0
```

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

とりあえず「ベース名. $\mbox{}$ 拡張子」と書いてください。

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

```
620 \xspcode'+=3
621 \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。

```
622 \xspcode'^^80=3
623 \xspcode'^^81=3
624 \xspcode'^^82=3
625 \xspcode'^^83=3
626 \xspcode'^^84=3
627 \xspcode'^^85=3
628 \xspcode'^^86=3
629 \xspcode'^^87=3
630 \xspcode'^^88=3
631 \xspcode'^^89=3
632 \xspcode'^^8a=3
633 \xspcode'^^8b=3
634 \xspcode'^^8c=3
635 \xspcode'^^8d=3
636 \xspcode'^^8e=3
637 \xspcode'^^8f=3
638 \xspcode'^^90=3
639 \xspcode'^^91=3
640 \xspcode'^^92=3
641 \xspcode'^^93=3
642 \xspcode'^^94=3
643 \xspcode'^^95=3
644 \xspcode'^^96=3
645 \xspcode'^^97=3
646 \xspcode'^^98=3
647 \xspcode'^^99=3
```

648 \xspcode'^9a=3
649 \xspcode'^9b=3
650 \xspcode'^9c=3
651 \xspcode'^9d=3
652 \xspcode'^9e=3
653 \xspcode'^9f=3
654 \xspcode'^a0=3
655 \xspcode'^a1=3
656 \xspcode'^a2=3
657 \xspcode'^a3=3
658 \xspcode'^a4=3
659 \xspcode'^a5=3
660 \xspcode'^a6=3
661 \xspcode'^a7=3
662 \xspcode'^a8=3
663 \xspcode'^a9=3
664 \xspcode'^aa=3
665 \xspcode'^ab=3
666 \xspcode'^ac=3
667 \xspcode'^ad=3
668 \xspcode'^ae=3
669 \xspcode'^af=3
670 \xspcode'^b0=3
671 \xspcode'^b1=3
672 \xspcode'^b2=3
673 \xspcode'^b3=3
674 \xspcode'^b4=3
675 \xspcode'^b5=3
676 \xspcode'^b6=3
677 \xspcode'^b7=3
678 \xspcode'^b8=3
679 \xspcode'^b9=3
680 \xspcode'^ba=3
681 \xspcode'^bb=3
682 \xspcode'^bc=3
683 \xspcode'^bd=3
684 \xspcode'^be=3
685 \xspcode'^bf=3
686 \xspcode'^c0=3
687 \xspcode'^c1=3
688 \xspcode'^c2=3
689 \xspcode'^c3=3
690 \xspcode'^c4=3
691 \xspcode'^c5=3
692 \xspcode'^c6=3
693 \xspcode'^c7=3
694 \xspcode'^c8=3
695 \xspcode'^c9=3
696 \xspcode'^ca=3

697 \xspcode'^^cb=3
698 \xspcode'^^cc=3
699 \xspcode'^^cd=3
700 \xspcode'^^ce=3
701 \xspcode'^^cf=3
702 \xspcode'^^d0=3
703 \xspcode'^^d1=3
704 \xspcode'^^d2=3
705 \xspcode'^^d3=3
706 \xspcode'^^d4=3
707 \xspcode'^^d5=3
708 \xspcode'^^d6=3
709 \xspcode'^^d7=3
710 \xspcode'^^d8=3
711 \xspcode'^^d9=3
712 \xspcode'^^da=3
713 \xspcode'^^db=3
714 \xspcode'^^dc=3
715 \xspcode'^^dd=3
716 \xspcode'^^de=3
717 \xspcode'^^df=3
718 \xspcode'^^e0=3
719 \xspcode'^^e1=3
720 \xspcode'^^e2=3
721 \xspcode'^^e3=3
722 \xspcode'^^e4=3
723 \xspcode'^^e5=3
724 \xspcode'^^e6=3
725 \xspcode'^^e7=3
726 \xspcode'^^e8=3
727 \xspcode'^^e9=3
728 \xspcode'^^ea=3
729 \xspcode'^^eb=3
730 \xspcode'^^ec=3
731 \xspcode'^^ed=3
732 \xspcode'^^ee=3
733 \xspcode'^^ef=3
734 \xspcode'^^f0=3
735 \xspcode'^^f1=3
736 \xspcode'^^f2=3
737 \xspcode'^^f3=3
738 \xspcode'^^f4=3
739 \xspcode'^^f5=3
740 \xspcode'^^f6=3
741 \xspcode'^^f7=3
742 \xspcode'^^f8=3
743 \xspcode'^^f9=3
744 \xspcode'^^fa=3
745 \xspcode'^^fb=3

```

746 \xspcode{^^fc=3
747 \xspcode{^^fd=3
748 \xspcode{^^fe=3
749 \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 $\text{\{}}$ を補いました。

```

750 \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	\@xipt	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。

```

751 \def\@setfontsize#1#2#3{%
752 % \@nomath#1%
753 \ifx\protect\@typeset@protect
754   \let\@currsz#1%
755   \fi
756   \fontsize{#2}{#3}\selectfont
757   \ifdim\parindent>\z@
758     \if@english
759       \parindent=1em
760     \else
761       \parindent=1zw
762     \fi
763   \fi
764   \kanjiskip=0zw plus .1zw minus .01zw
765   \ifdim\xkanjiskip>\z@
766     \if@slide \xkanjiskip=0.1em \else
767       \xkanjiskip=0.25em plus 0.15em minus 0.06em
768     \fi
769   \fi}

```

`\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。

```

770 \def\jsc@setfontsize#1#2#3{%
771   \@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。

```

772 \emergencystretch 3zw

```

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

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

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

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

```

773 \newif\ifnarrowbaselines
774 \if@english
775   \narrowbaselinestrue
776 \fi
777 \def\narrowbaselines{%
778   \narrowbaselinestrue
779   \skip0=\abovedisplayskip
780   \skip2=\abovedisplayshortskip
781   \skip4=\belowdisplayskip
782   \skip6=\belowdisplayshortskip
783   \@currsz\selectfont
784   \abovedisplayskip=\skip0

```

```

785 \abovedisplayskip=\skip2
786 \belowdisplayskip=\skip4
787 \belowdisplayskip=\skip6\relax}
788 \def\widebaselines{\narrowbaselinesfalse\@currsize\selectfont}

```

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

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

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

```

789 \renewcommand{\normalsize}{%
790   \ifnarrowbaselines
791     \jsc@setfontsize\normalsize\@xpt\@xiipt
792   \else
793     \jsc@setfontsize\normalsize\@xpt{\n@baseline}%
794   \fi

```

数式の上のアキ (`\abovedisplayskip`)、短い数式の上のアキ (`\abovedisplayskipshortskip`)、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] \TeX Q & A 52569 から始まる議論について逡巡していましたが、結局、微調節してみることにしました。

```

795 \abovedisplayskip 11\jsc@empt \@plus3\jsc@empt \@minus4\jsc@empt
796 \abovedisplayskipshortskip \z@ \@plus3\jsc@empt
797 \belowdisplayskip 9\jsc@empt \@plus3\jsc@empt \@minus4\jsc@empt
798 \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。

```

799 \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。

```

800 \normalsize

```

`\Cht` 基準となる長さの設定をします。p \LaTeX 2 ϵ カーネル (`plfonts.dtx`) で宣言されているパ

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

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

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

```

801 \setbox0\hbox{\char\jis"3441}%
802 \setlength\Cht{\ht0}
803 \setlength\Cdp{\dp0}
804 \setlength\Cwd{\wd0}
805 \setlength\Cvs{\baselineskip}

```

```
806 \setlength\Chs{\wd0}
807 \setbox0=\box\voidb@x
```

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

```
808 \newcommand{\small}{%
809   \ifnarrowbaselines
810 (!kiyou)   \jsc@setfontsize\small\@ixpt{11}%
811 (kiyou)   \jsc@setfontsize\small{8.8888}{11}%
812   \else
813 (!kiyou)   \jsc@setfontsize\small\@ixpt{13}%
814 (kiyou)   \jsc@setfontsize\small{8.8888}{13.2418}%
815   \fi
816 \abovedisplayskip 9\jsc@mpt \@plus3\jsc@mpt \@minus4\jsc@mpt
817 \abovedisplayshortskip \z@ \@plus3\jsc@mpt
818 \belowdisplayskip \abovedisplayskip
819 \belowdisplayshortskip \belowdisplayskip
820 \def\@listi{\leftmargin\leftmargini
821           \topsep \z@
822           \parsep \z@
823           \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。

```
824 \newcommand{\footnotesize}{%
825   \ifnarrowbaselines
826 (!kiyou)   \jsc@setfontsize\footnotesize\@viiipt{9.5}%
827 (kiyou)   \jsc@setfontsize\footnotesize{8.8888}{11}%
828   \else
829 (!kiyou)   \jsc@setfontsize\footnotesize\@viiipt{11}%
830 (kiyou)   \jsc@setfontsize\footnotesize{8.8888}{13.2418}%
831   \fi
832 \abovedisplayskip 6\jsc@mpt \@plus2\jsc@mpt \@minus3\jsc@mpt
833 \abovedisplayshortskip \z@ \@plus2\jsc@mpt
834 \belowdisplayskip \abovedisplayskip
835 \belowdisplayshortskip \belowdisplayskip
836 \def\@listi{\leftmargin\leftmargini
837           \topsep \z@
838           \parsep \z@
839           \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
`\tiny` 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
`\large` 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
`\Large` 間で行が揃うようにします。

```
\LARGE
\huge
\Huge
\HUGE
```

[2004-11-03] \HUGE を追加。

```
840 \newcommand{\scriptsize}{\jsc@setfontsize\scriptsize\@viipt\@viipt}
841 \newcommand{\tiny}{\jsc@setfontsize\tiny\@vpt\@vpt}
842 \if@twocolumn
843 (!kiyou) \newcommand{\large}{\jsc@setfontsize\large\@xiipt{\n@baseline}}
844 (kiyou) \newcommand{\large}{\jsc@setfontsize\large{11.111}{\n@baseline}}
845 \else
846 (!kiyou) \newcommand{\large}{\jsc@setfontsize\large\@xiipt{17}}
847 (kiyou) \newcommand{\large}{\jsc@setfontsize\large{11.111}{17}}
848 \fi
849 (!kiyou)\newcommand{\Large}{\jsc@setfontsize\Large\@xivpt{21}}
850 (kiyou)\newcommand{\Large}{\jsc@setfontsize\Large{12.222}{21}}
851 \newcommand{\LARGE}{\jsc@setfontsize\LARGE\@xviipt{25}}
852 \newcommand{\huge}{\jsc@setfontsize\huge\@xxpt{28}}
853 \newcommand{\Huge}{\jsc@setfontsize\Huge\@xxvpt{33}}
854 \newcommand{\HUGE}{\jsc@setfontsize\HUGE{30}{40}}
```

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

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

```
855 \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。『`LaTeXε` 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。

```
856 % \newcommand{\headfont}{\bfseries}
857 \newcommand{\headfont}{\gtfamily\sffamily}
858 % \newcommand{\headfont}{\sffamily\fontseries{sbc}\selectfont}
```

5 レイアウト

■二段組

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

```
859 (!kiyou)\setlength\columnsep{2zw}
860 (kiyou)\setlength\columnsep{28truebp}
861 \setlength\columnseprule{\z@}
```

■段落

`\lineskip` 上下の行の文字が `\lineskiplimit` より接近したら、`\lineskip` より近づかないようにします。元は 0pt ですが 1pt に変更しました。normal... の付いた方は保存用です。

`\normallineskip`

`\lineskiplimit` 862 `\setlength\lineskip{1\jsc@empt}`

`\normallineskiplimit` 863 `\setlength\normallineskip{1\jsc@empt}`

864 `\setlength\lineskiplimit{1\jsc@empt}`

865 `\setlength\normallineskiplimit{1\jsc@empt}`

`\baselinestretch` 実際の行送りが `\baselineskip` の何倍かを表すマクロです。たとえば

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

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

866 `\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。

```
867 \setlength\parskip{\z@}
868 \if@slide
869 \setlength\parindent{0zw}
870 \else
871 \setlength\parindent{1zw}
872 \fi
```

`\@lowpenalty` `\nopagebreak`、`\nolinebreak` は引数に応じて次のペナルティ値のうちどれかを選ぶよう

`\@medpenalty` になっています。ここはオリジナル通りです。

`\@highpenalty` 873 `\@lowpenalty 51`

874 `\@medpenalty 151`

875 `\@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。

```
876 % \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。

```
877 % \brokenpenalty 100
```

5.1 ページレイアウト

■縦方向のスペース

`\headheight` `\topskip` は本文領域上端と本文 1 行目のベースラインとの距離です。あまりぎりぎりの値にすると、本文中に f 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。

```
878 \setlength\topskip{1.38zw}%% from 10\jsc@mpt (2016-08-17)
879 \if@slide
880   \setlength\headheight{0\jsc@mpt}
881 \else
882   \setlength\headheight{20\jsc@mpt}%% from 2\topskip (2016-08-17); from \topskip (2003-06-26)
883 \fi
```

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

```
884 (*article | kiyou)
885 \if@slide
886   \setlength\footskip{0pt}
887 \else
888   \setlength\footskip{0.03367\paperheight}
889   \ifdim\footskip<\baselineskip
890     \setlength\footskip{\baselineskip}
891   \fi
892 \fi
893 (/article | kiyou)
894 (jspf)\setlength\footskip{9\jsc@mmm}
895 (*book)
896 \if@report
897   \setlength\footskip{0.03367\paperheight}
898   \ifdim\footskip<\baselineskip
899     \setlength\footskip{\baselineskip}
900   \fi
901 \else
902   \setlength\footskip{0pt}
903 \fi
904 (/book)
905 (*report)
906 \setlength\footskip{0.03367\paperheight}
907 \ifdim\footskip<\baselineskip
908   \setlength\footskip{\baselineskip}
909 \fi
910 (/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)。

```
911 <*article>
912 \if@slide
913   \setlength\headsep{0\jsc@empt}
914   \addtolength\headsep{-\topskip}%% added (2016-10-08)
915   \addtolength\headsep{10\jsc@empt}%% added (2016-10-08)
916 \else
917   \setlength\headsep{\footskip}
918   \addtolength\headsep{-\topskip}
919 \fi
920 </article>
921 <*book>
922 \if@report
923   \setlength\headsep{\footskip}
924   \addtolength\headsep{-\topskip}
925 \else
926   \setlength\headsep{6\jsc@mmm}
927   \addtolength\headsep{-\topskip}%% added (2016-10-08)
928   \addtolength\headsep{10\jsc@empt}%% added (2016-10-08)
929 \fi
930 </book>
931 <*report>
932 \setlength\headsep{\footskip}
933 \addtolength\headsep{-\topskip}
934 </report>
935 <*jspf>
936 \setlength\headsep{9\jsc@mmm}
937 \addtolength\headsep{-\topskip}
938 </jspf>
939 <*kiyou>
940 \setlength\headheight{0\jsc@empt}
941 \setlength\headsep{0\jsc@empt}
942 \addtolength\headsep{-\topskip}%% added (2016-10-08)
943 \addtolength\headsep{10\jsc@empt}%% added (2016-10-08)
944 </kiyou>
```

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

```
945 \setlength\maxdepth{.5\topskip}
```

■本文の幅と高さ

`\fullwidth` 本文の幅が全角 40 文字を超えると読みにくくなります。そこで、書籍の場合に限って、紙

の幅が広いときは外側のマージンを余分にとって全角 40 文字に押え、ヘッダやフッタは本文領域より広く取ることにします。このときヘッダやフッタの幅を表す `\fullwidth` という長さを定義します。

```
946 \newdimen\fullwidth
```

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

`\textwidth` 書籍以外では本文領域の幅 `\textwidth` は `\fullwidth` と等しくします。`article` では A4 縦置きで 49 文字となります。某学会誌スタイルでは 50zw (25 文字× 2 段) + 段間 8mm とします。

```
947 (*article)
948 \if@slide
949   \setlength\fullwidth{0.9\paperwidth}
950 \else
951   \setlength\fullwidth{0.76\paperwidth}
952 \fi
953 \if@twocolumn \@tempdima=2zw \else \@tempdima=1zw \fi
954 \divide\fullwidth\@tempdima \multiply\fullwidth\@tempdima
955 \setlength\textwidth{\fullwidth}
956 (/article)
957 (*book)
958 \if@report
959   \setlength\fullwidth{0.76\paperwidth}
960 \else
961   \setlength\fullwidth{\paperwidth}
962   \addtolength\fullwidth{-36\jsc@mmm}
963 \fi
964 \if@twocolumn \@tempdima=2zw \else \@tempdima=1zw \fi
965 \divide\fullwidth\@tempdima \multiply\fullwidth\@tempdima
966 \setlength\textwidth{\fullwidth}
967 \if@report \else
968   \if@twocolumn \else
969     \ifdim \fullwidth>40zw
970       \setlength\textwidth{40zw}
971     \fi
972   \fi
973 \fi
974 (/book)
975 (*report)
976 \setlength\fullwidth{0.76\paperwidth}
977 \if@twocolumn \@tempdima=2zw \else \@tempdima=1zw \fi
978 \divide\fullwidth\@tempdima \multiply\fullwidth\@tempdima
979 \setlength\textwidth{\fullwidth}
980 (/report)
981 (*jspf)
```

```

982 \setlength\fullwidth{50zw}
983 \addtolength\fullwidth{8\jsc@mmm}
984 \setlength\textwidth{\fullwidth}
985 \jpsf)
986 (*kiyou)
987 \setlength\fullwidth{48zw}
988 \addtolength\fullwidth{\columnsep}
989 \setlength\textwidth{\fullwidth}
990 \jpsf)

```

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

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

某学会誌スタイルでは 44 行にします。

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

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

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

```

991 (*article | book | report)
992 \if@slide
993   \setlength{\textheight}{0.95\paperheight}
994 \else
995   \setlength{\textheight}{0.83\paperheight}
996 \fi
997 \addtolength{\textheight}{-10\jsc@mpt}%% from -\topskip (2016-10-08); from -\headheight (2003-
998 \addtolength{\textheight}{-\headsep}
999 \addtolength{\textheight}{-\footskip}
1000 \addtolength{\textheight}{-\topskip}
1001 \divide\textheight\baselineskip
1002 \multiply\textheight\baselineskip
1003 \jpsf)
1004 \jpsf)\setlength{\textheight}{51\baselineskip}
1005 (kiyou)\setlength{\textheight}{47\baselineskip}
1006 \addtolength{\textheight}{\topskip}
1007 \addtolength{\textheight}{0.1\jsc@mpt}
1008 \jpsf)\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 $\text{\LaTeX} 2_{\epsilon}$ 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。

```
1009 \def\flushbottom{%
1010   \def\@textbottom{\vskip \z@ \@plus.1\jcs@empt}%
1011   \let\@texttop\relax}
```

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

```
1012 \setlength\marginparsep{\columnsep}
1013 \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` が使われます。TEX は上・左マージンに `1truein` を挿入しますが、トンボ関係のオプションが指定されると pLATEX 2_ε (`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。

```
1014 \setlength{\oddsidemargin}{\paperwidth}
1015 \addtolength{\oddsidemargin}{-\fullwidth}
1016 \setlength{\oddsidemargin}{.5\oddsidemargin}
1017 \iftombow
1018   \addtolength{\oddsidemargin}{-1in}
1019 \else
1020   \addtolength{\oddsidemargin}{-\inv@mag in}
1021 \fi
1022 \setlength{\evensidemargin}{\oddsidemargin}
1023 \if@mparswitch
1024   \addtolength{\evensidemargin}{\fullwidth}
1025   \addtolength{\evensidemargin}{-\textwidth}
1026 \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。

```
1027 \setlength\marginparwidth{\paperwidth}
1028 \addtolength\marginparwidth{-\oddsidemargin}
1029 \addtolength\marginparwidth{-\inv@mag in}
1030 \addtolength\marginparwidth{-\textwidth}
1031 \addtolength\marginparwidth{-10\jcs@mmm}
1032 \addtolength\marginparwidth{-\marginparsep}
1033 \@tempdima=1zw
1034 \divide\marginparwidth\@tempdima
1035 \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)。

```
1036 \setlength\topmargin{\paperheight}
1037 \addtolength\topmargin{-\textheight}
1038 \if@slide
1039   \addtolength\topmargin{-\headheight}
1040 \else
1041   \addtolength\topmargin{-10\jsc@empt}%% from -\topskip (2016-10-08); from -\headheight (2003-0
1042 \fi
1043 \addtolength\topmargin{-\headsep}
1044 \addtolength\topmargin{-\footskip}
1045 \setlength\topmargin{0.5\topmargin}
1046 (kiyou)\setlength\topmargin{81truebp}
1047 \iftombow
1048   \addtolength\topmargin{-1in}
1049 \else
1050   \addtolength\topmargin{-\inv@mag in}
1051 \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。

```
1052 {\footnotesize\global\setlength\footnotesep{\baselineskip}}
1053 \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。

```
1054 \setlength{\skip\footins}{16\jsc@empt \@plus 5\jsc@empt \@minus 2\jsc@empt}
```

■フロート関連 フロート (図, 表) 関連のパラメータは L^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。

```
1055 \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。

```
1056 \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。

```
1057 \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。

```
1058 \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。

```
1059 \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。

```
1060 \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。

```
1061 \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。

```
1062 \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。

```
1063 \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。

```
1064 \renewcommand{\dblfloatpagefraction}{.8}
```

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

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

```
1065 \setlength\floatsep {12\jsc@mpt \@plus 2\jsc@mpt \@minus 2\jsc@mpt}
1066 \setlength\textfloatsep{20\jsc@mpt \@plus 2\jsc@mpt \@minus 4\jsc@mpt}
1067 \setlength\intextsep {12\jsc@mpt \@plus 2\jsc@mpt \@minus 2\jsc@mpt}
```

`\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 1068 \setlength\dblfloatsep {12\jsc@mpt \@plus 2\jsc@mpt \@minus 2\jsc@mpt}
1069 \setlength\dbltextfloatsep{20\jsc@mpt \@plus 2\jsc@mpt \@minus 4\jsc@mpt}
```

`\@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 1070 \setlength\@fptop{0\jsc@mpt \@plus 1fil}
1071 \setlength\@fpsep{8\jsc@mpt \@plus 2fil}
1072 \setlength\@fpbot{0\jsc@mpt \@plus 1fil}
```

`\@dblftop` 段抜きフロートについての値です。

`\@dblfpsep` 1073 `\setlength\@dblftop{0\jsc@empt \@plus 1fil}`

`\@dblfpbot` 1074 `\setlength\@dblfpsep{8\jsc@empt \@plus 2fil}`

1075 `\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 p_ET_EX の標準クラス 2017/02/15 に合わせて、同じ命令を追加しました。

- `\pltx@cleartooddpage` 1. `\pltx@cleartorightpage` : 右ページになるまでページを繰る命令
- `\pltx@cleartoevenpage` 2. `\pltx@cleartoleftpage` : 左ページになるまでページを繰る命令
3. `\pltx@cleartooddpage` : 奇数ページになるまでページを繰る命令
4. `\pltx@cleartoevenpage` : 偶数ページになるまでページを繰る命令

となっています。

```
1076 (*article | book | report)
1077 \def\pltx@cleartorightpage{\clearpage\if@twoside
1078   \ifodd\c@page
1079     \iftdir
1080       \hbox{}\thispagestyle{empty}\newpage
1081       \if@twocolumn\hbox{}\newpage\fi
1082     \fi
1083   \else
1084     \ifydir
1085       \hbox{}\thispagestyle{empty}\newpage
1086       \if@twocolumn\hbox{}\newpage\fi
1087     \fi
1088   \fi\fi}
1089 \def\pltx@cleartoleftpage{\clearpage\if@twoside
1090   \ifodd\c@page
1091     \ifydir
1092       \hbox{}\thispagestyle{empty}\newpage
1093       \if@twocolumn\hbox{}\newpage\fi
1094     \fi
1095   \else
1096     \iftdir
1097       \hbox{}\thispagestyle{empty}\newpage
1098       \if@twocolumn\hbox{}\newpage\fi
1099     \fi
1100   \fi\fi}
1101 \def\pltx@cleartooddpage{\clearpage\if@twoside
1102   \ifodd\c@page\else
1103     \hbox{}\thispagestyle{empty}\newpage
1104     \if@twocolumn\hbox{}\newpage\fi
1105   \fi\fi}
1106 \def\pltx@cleartoevenpage{\clearpage\if@twoside
```

```

1107 \ifodd\c@page
1108   \hbox{\thispagestyle{empty}}\newpage
1109   \if@twocolumn\hbox{\newpage}\fi
1110   \fi\fi}
1111 </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。

```

1112 (*book | report)
1113 \if@openleft
1114   \let\cleardoublepage\ptx@cleartoleftpage
1115 \else\if@openright
1116   \let\cleardoublepage\ptx@cleartorightpage
1117 \fi\fi
1118 </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 (ヘッド, フッタ) を出力する命令です。これらは `\fullwidth` 幅の `\hbox` の中で呼び出されます。
`\@evenfoot` `\ps@...` の中で定義しておきます。

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

```

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

```

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

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

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

```

1119 % \def\ps@empty{%
1120 %   \let\@mkboth\@gobbletwo
1121 %   \let\@oddhead\@empty
1122 %   \let\@oddfoot\@empty
1123 %   \let\@evenhead\@empty
1124 %   \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。

```

1125 \def\ps@plainfoot{%
1126   \let\@mkboth\@gobbletwo
1127   \let\@oddhead\@empty
1128   \def\@oddfoot{\normalfont\hfil\thepage\hfil}%
1129   \let\@evenhead\@empty
1130   \let\@evenfoot\@oddfoot}
1131 \def\ps@plainhead{%
1132   \let\@mkboth\@gobbletwo
1133   \let\@oddfoot\@empty
1134   \let\@evenfoot\@empty
1135   \def\@evenhead{%
1136     \if@mparswitch \hss \fi
1137     \hbox to \fullwidth{\textbf{\thepage}\hfil}%
1138     \if@mparswitch\else \hss \fi}%
1139   \def\@oddhead{%
1140     \hbox to \fullwidth{\hfil\textbf{\thepage}}\hss}}
1141 \langle book \rangle \if@report \let\ps@plain\ps@plainfoot \else \let\ps@plain\ps@plainhead \fi
1142 \langle !book \rangle \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。

```

1143 \langle *article | kiyou \rangle
1144 \if@twoside
1145   \def\ps@headings{%
1146     \let\@oddfoot\@empty
1147     \let\@evenfoot\@empty
1148     \def\@evenhead{\if@mparswitch \hss \fi
1149       \underline{\hbox to \fullwidth{\textbf{\thepage}\hfil\leftmark}}}%
1150     \if@mparswitch\else \hss \fi}%
1151     \def\@oddhead{%
1152       \underline{%
1153         \hbox to \fullwidth{\rightmark}\hfil\textbf{\thepage}}\hss}%
1154     \let\@mkboth\markboth
1155     \def\sectionmark##1{\markboth{%
1156       \ifnum \c@secnumdepth >\z@ \thesection \hskip1zw\fi
1157       ##1}\fi}%
1158     \def\subsectionmark##1{\markright{%

```

```

1159     \ifnum \c@secnumdepth >\@ne \thesubsection \hskip1zw\fi
1160     ##1}}%
1161 }
1162 \else % if not twoside
1163 \def\ps@headings{%
1164 \let\@oddfont\@empty
1165 \def\@oddhead{%
1166 \underline{%
1167 \hbox to \fullwidth{\rightmark}\hfil\textbf{\thepage}}\hss}%
1168 \let\@mkboth\markboth
1169 \def\sectionmark##1{\markright{%
1170 \ifnum \c@secnumdepth >\z@ \thesection \hskip1zw\fi
1171 ##1}}
1172 \fi
1173 </article | kiyou)

```

次は book および report の場合です。[2011-05-10] しっぽ愛好家さん [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）。

```

1174 (*book | report)
1175 \newif\if@omit@number
1176 \def\ps@headings{%
1177 \let\@oddfont\@empty
1178 \let\@evenfont\@empty
1179 \def\@evenhead{%
1180 \if@mparswitch \hss \fi
1181 \underline{\hbox to \fullwidth{\autoxspacing
1182 \textbf{\thepage}\hfil\leftmark}}%
1183 \if@mparswitch\else \hss \fi}%
1184 \def\@oddhead{\underline{\hbox to \fullwidth{\autoxspacing
1185 {\if@twoside\rightmark\else\leftmark\fi}\hfil\textbf{\thepage}}}\hss}%
1186 \let\@mkboth\markboth
1187 \def\chaptermark##1{\markboth{%
1188 \ifnum \c@secnumdepth >\@m@ne
1189 (book) \if@mainmatter
1190 \if@omit@number\else
1191 \@chapapp\thechapter\@chappos\hskip1zw
1192 \fi
1193 (book) \fi
1194 \fi
1195 ##1}{}}%
1196 \def\sectionmark##1{\markright{%
1197 \ifnum \c@secnumdepth >\z@ \thesection \hskip1zw\fi
1198 ##1}}%
1199 </book | report)

```

最後は学会誌の場合です。

```

1200 (*jspf)
1201 \def\ps@headings{%
1202 \def\@oddfont{\normalfont\hfil\thepage\hfil}

```

```

1203 \def\@evenfoot{\normalfont\hfil\thepage\hfil}
1204 \def\@oddhead{\normalfont\hfil \@title \hfil}
1205 \def\@evenhead{\normalfont\hfil プラズマ・核融合学会誌 \hfil}}
1206 </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。

```

1207 \def\ps@myheadings{%
1208 \let\@oddfont\@empty\let\@evenfont\@empty
1209 \def\@evenhead{%
1210 \if@mparswitch \hss \fi%
1211 \hbox to \fullwidth{\thepage\hfil\leftmark}%
1212 \if@mparswitch\else \hss \fi}%
1213 \def\@oddhead{%
1214 \hbox to \fullwidth{\rightmark\hfil\thepage}\hss}%
1215 \let\@mkboth\@gobbletwo
1216 (book|report) \let\chaptermark\@gobble
1217 \let\sectionmark\@gobble
1218 (!book&!report) \let\subsectionmark\@gobble
1219 }

```

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 1220 % \newcommand*\title}[1]{\gdef\@title{#1}}
1221 % \newcommand*\author}[1]{\gdef\@author{#1}}
1222 % \newcommand*\date}[1]{\gdef\@date{#1}}
1223 % \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 1224 <*jspf>
1225 \newcommand*\etitle}[1]{\gdef\@etitle{#1}}
\keywords 1226 \newcommand*\eauthor}[1]{\gdef\@eauthor{#1}}
1227 \newcommand*\keywords}[1]{\gdef\@keywords{#1}}
1228 \newcommand*\email}[1]{\gdef\@authors@mail{#1}}
1229 \newcommand*\AuthorsEmail}[1]{\gdef\@authors@mail{author's e-mail:\ #1}}
1230 </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

命を作ることにします。

```
1231 \def\plainifnotempty{%
1232   \ifx \@oddhead \@empty
1233     \ifx \@oddfoot \@empty
1234     \else
1235       \thispagestyle{plainfoot}%
1236     \fi
1237   \else
1238     \thispagestyle{plainhead}%
1239   \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。

```
1240 (*article | book | report | kiyou)
1241 \if@titlepage
1242   \newcommand{\maketitle}{%
1243     \begin{titlepage}%
1244       \let\footnotesize\small
1245       \let\footnoterule\relax
1246       \let\footnote\thanks
1247       \null\vfil
1248       \if@slide
1249         {\footnotesize \@date}%
1250       \begin{center}
1251         \mbox{} \\\[1zw]
1252         \large
1253         {\maybeblue\hrule height0\jsc@mpt depth2\jsc@mpt\relax}\par
1254         \jsc@smallskip
1255         \@title
1256         \jsc@smallskip
1257         {\maybeblue\hrule height0\jsc@mpt depth2\jsc@mpt\relax}\par
1258         \vfill
1259         {\small \@author}%
1260       \end{center}
1261     \else
1262     \vskip 60\jsc@mpt
1263     \begin{center}%
1264       {\LARGE \@title \par}%
1265       \vskip 3em%
1266       {\large
1267         \lineskip .75em
1268         \begin{tabular}[t]{c}%
1269           \@author
1270         \end{tabular}\par}%
1271       \vskip 1.5em
```

```

1272     {\large \@date \par}%
1273     \end{center}%
1274     \fi
1275     \par
1276     \@thanks\vfil\null
1277 \end{titlepage}%
1278 \setcounter{footnote}{0}%
1279 \global\let\thanks\relax
1280 \global\let\maketitle\relax
1281 \global\let\@thanks\@empty
1282 \global\let\@author\@empty
1283 \global\let\@date\@empty
1284 \global\let\@title\@empty
1285 \global\let\title\relax
1286 \global\let\author\relax
1287 \global\let\date\relax
1288 \global\let\and\relax
1289 }%
1290 \else
1291 \newcommand{\maketitle}{\par
1292 \begingroup
1293 \renewcommand\thefootnote{\@fnsymbol\c@footnote}%
1294 \def\@makefnmark{\rlap{\@textsuperscript{\normalfont\@thefnmark}}}%
1295 \long\def\@makefntext##1{\advance\leftskip 3zw
1296 \parindent 1zw\noindent
1297 \llap{\@textsuperscript{\normalfont\@thefnmark}\hskip0.3zw}##1}%
1298 \if@twocolumn
1299 \ifnum \col@number=\@ne
1300 \maketitle
1301 \else
1302 \twocolumn[\maketitle]%
1303 \fi
1304 \else
1305 \newpage
1306 \global\@topnum\z@ % Prevents figures from going at top of page.
1307 \maketitle
1308 \fi
1309 \plainifnotempty
1310 \@thanks
1311 \endgroup
1312 \setcounter{footnote}{0}%
1313 \global\let\thanks\relax
1314 \global\let\maketitle\relax
1315 \global\let\@thanks\@empty
1316 \global\let\@author\@empty
1317 \global\let\@date\@empty
1318 \global\let\@title\@empty
1319 \global\let\title\relax
1320 \global\let\author\relax

```

```

1321 \global\let\date\relax
1322 \global\let\and\relax
1323 }

```

`\@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。

```

1324 \def\@maketitle{%
1325 \newpage\null
1326 \vskip 2em
1327 \begin{center}%
1328 \let\footnote\thanks
1329 {\LARGE \@title \par}%
1330 \vskip 1.5em
1331 {\large
1332 \lineskip .5em
1333 \begin{tabular}[t]{c}%
1334 \@author
1335 \end{tabular}\par}%
1336 \vskip 1em
1337 {\large \@date}%
1338 \end{center}%
1339 \par\vskip 1.5em
1340 (article | report | kiyou) \ifvoid\@abstractbox\else\centerline{\box\@abstractbox}\vskip1.5em\fi
1341 }
1342 \fi
1343 </article | book | report | kiyou>
1344 <*jspf>
1345 \newcommand{\maketitle}{\par
1346 \begingroup
1347 \renewcommand\thefootnote{\@fnsymbol\c@footnote}%
1348 \def\@makefnmark{\rlap{\@textsuperscript{\normalfont\@thefnmark}}}%
1349 \long\def\@makefntext##1{\advance\leftskip 3zw
1350 \parindent 1zw\noindent
1351 \llap{\@textsuperscript{\normalfont\@thefnmark}\hskip0.3zw}##1}%
1352 \twocolumn[\@maketitle]%
1353 \plainifnotempty
1354 \@thanks
1355 \endgroup
1356 \setcounter{footnote}{0}%
1357 \global\let\thanks\relax
1358 \global\let\maketitle\relax
1359 \global\let\@thanks\@empty
1360 \global\let\@author\@empty
1361 \global\let\@date\@empty
1362 % \global\let\@title\@empty % \@title は柱に使う
1363 \global\let\title\relax
1364 \global\let\author\relax
1365 \global\let\date\relax
1366 \global\let\and\relax
1367 \ifx\authors@mail\@undefined\else{%

```

```

1368 \def\@makefntext{\advance\leftskip 3zw \parindent -3zw}%
1369 \footnotetext[0]{\itshape\authors@mail}%
1370 }\fi
1371 \global\let\authors@mail\@undefined}
1372 \def\@maketitle{%
1373 \newpage\null
1374 \vskip 6em % used to be 2em
1375 \begin{center}
1376 \let\footnote\thanks
1377 \ifx\@title\@undefined\else{\LARGE\headfont\@title\par}\fi
1378 \lineskip .5em
1379 \ifx\@author\@undefined\else
1380 \vskip 1em
1381 \begin{tabular}[t]{c}%
1382 \@author
1383 \end{tabular}\par
1384 \fi
1385 \ifx\@etitle\@undefined\else
1386 \vskip 1em
1387 {\large \@etitle \par}%
1388 \fi
1389 \ifx\@eauthor\@undefined\else
1390 \vskip 1em
1391 \begin{tabular}[t]{c}%
1392 \@eauthor
1393 \end{tabular}\par
1394 \fi
1395 \vskip 1em
1396 \@date
1397 \end{center}
1398 \vskip 1.5em
1399 \centerline{\box\@abstractbox}
1400 \ifx\@keywords\@undefined\else
1401 \vskip 1.5em
1402 \centerline{\parbox{157\jsc@mmm}{\textsf{Keywords:}}\ \small\@keywords}}
1403 \fi
1404 \vskip 1.5em}
1405 \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。

```

1406 \def\@startsection#1#2#3#4#5#6{%
1407   \if@noskipsec \leavevmode \fi
1408   \par
1409 % 見出し上の空きを \@tempskipa にセットする
1410   \@tempskipa #4\relax
1411 % \@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
1412   \if@english \@afterindentfalse \else \@afterindenttrue \fi
1413 % 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
1414   \ifdim \@tempskipa <\z@
1415     \@tempskipa -\@tempskipa \@afterindentfalse
1416   \fi
1417   \if@nobreak
1418 %   \everypar{\everyparhook}% これは間違い
1419     \everypar{}%
1420   \else
1421     \addpenalty\@secpenalty
1422 % 次の行は削除
1423 %   \addvspace\@tempskipa
1424 % 次の \noindent まで追加
1425     \ifdim \@tempskipa >\z@
1426       \if@slide\else
1427         \null
1428         \vspace*{-\baselineskip}%
1429       \fi
1430       \vskip\@tempskipa
1431     \fi
1432   \fi
1433   \noindent

```

```

1434 % 追加終わり
1435 \@ifstar
1436     {\@ssect{#3}{#4}{#5}{#6}}%
1437     {\@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。
1438 \def\@sect#1#2#3#4#5#6[#7]#8{%
1439     \ifnum #2>\c@secnumdepth
1440         \let\@svsec\@empty
1441     \else
1442         \refstepcounter{#1}%
1443         \protected@edef\@svsec{\@secntformat{#1}\relax}%
1444     \fi
1445 % 見出し後の空きを \@tempskipa にセット
1446     \@tempskipa #5\relax
1447 % 条件判断の順序を入れ替えました
1448     \ifdim \@tempskipa<\z@
1449         \def\@svsechd{%
1450             #6{\hskip #3\relax
1451                 \@svsec #8}%
1452             \csname #1mark\endcsname{#7}}%
1453         \addcontentsline{toc}{#1}{%
1454             \ifnum #2>\c@secnumdepth \else
1455                 \protect\numberline{\csname the#1\endcsname}%
1456             \fi
1457             #7}}% 目次にフルネームを載せるなら #8
1458     \else
1459         \begingroup
1460             \interlinepenalty \@M % 下から移動
1461             #6{%
1462                 \@hangfrom{\hskip #3\relax\@svsec}%
1463             \interlinepenalty \@M % 上に移動
1464             #8\@@par}%
1465         \endgroup
1466         \csname #1mark\endcsname{#7}}%
1467     \addcontentsline{toc}{#1}{%
1468         \ifnum #2>\c@secnumdepth \else
1469             \protect\numberline{\csname the#1\endcsname}%
1470         \fi
1471         #7}}% 目次にフルネームを載せるならここは #8
1472     \fi
1473     \@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。

```

1474 \def\@xsect#1{%

```

```

1475 % 見出しの後ろの空きを \@tempskipa にセット
1476 \@tempskipa #1\relax
1477 % 条件判断の順序を変えました
1478 \ifdim \@tempskipa<\z@
1479   \@nobreakfalse
1480   \global\@noskipsectrue
1481   \everypar{%
1482     \if@noskipsec
1483       \global\@noskipsecfalse
1484       {\setbox\z@\lastbox}%
1485       \clubpenalty\@M
1486       \begingroup \svsechd \endgroup
1487       \unskip
1488       \@tempskipa #1\relax
1489       \hskip -\@tempskipa
1490     \else
1491       \clubpenalty \@clubpenalty
1492       \everypar{\everyparhook}%
1493       \fi\everyparhook}%
1494   \else
1495     \par \nobreak
1496     \vskip \@tempskipa
1497     \@afterheading
1498   \fi
1499   \if@slide
1500     {\vskip\if@twocolumn-5\jsc@empt\else-6\jsc@empt\fi
1501     \maybeblue\hrule height0\jsc@empt depth1\jsc@empt
1502     \vskip\if@twocolumn 4\jsc@empt\else 7\jsc@empt\fi\relax}%
1503   \fi
1504   \par % 2000-12-18
1505   \ignorespaces}
1506 \def\@ssect#1#2#3#4#5{%
1507   \@tempskipa #3\relax
1508   \ifdim \@tempskipa<\z@
1509     \def\@svsechd{#4{\hskip #1\relax #5}}%
1510   \else
1511     \begingroup
1512     #4{%
1513       \@hangfrom{\hskip #1}%
1514       \interlinepenalty \@M #5\@@par}%
1515     \endgroup
1516   \fi
1517   \@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` ¹⁵¹⁸ `\newcommand*\chaptermark[1]{}`

`\subsubsectionmark`

`\paragraphmark`

`\subparagraphmark`

```

1519 % \newcommand*\sectionmark}[1]{ }
1520 % \newcommand*\subsectionmark}[1]{ }
1521 % \newcommand*\subsubsectionmark}[1]{ }
1522 % \newcommand*\paragraphmark}[1]{ }
1523 % \newcommand*\subparagraphmark}[1]{ }

```

■カウンタの定義

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

```

1524 (!book&!report)\setcounter{secnumdepth}{3}
1525 (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 1526 \newcounter{part}
1527 (book | report)\newcounter{chapter}
\c@subsubsection 1528 (book | report)\newcounter{section}[chapter]
\c@paragraph 1529 (!book&!report)\newcounter{section}
\c@subparagraph 1530 \newcounter{subsection}[section]
1531 \newcounter{subsubsection}[subsection]
1532 \newcounter{paragraph}[subsubsection]
1533 \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。

```

1534 \renewcommand{\thepart}{\@Roman\c@part}
1535 (!book&!report)% \renewcommand{\thesection}{\@arabic\c@section}
1536 (!book&!report)\renewcommand{\thesection}{\presectionname\@arabic\c@section\postsectionname}
1537 (!book&!report)\renewcommand{\thesubsection}{\@arabic\c@section.\@arabic\c@subsection}
1538 (*book | report)
1539 \renewcommand{\thechapter}{\@arabic\c@chapter}
1540 \renewcommand{\thesection}{\thechapter.\@arabic\c@section}
1541 \renewcommand{\thesubsection}{\thesection.\@arabic\c@subsection}
1542 (/book | report)
1543 \renewcommand{\thesubsubsection}{%
1544   \thesubsection.\@arabic\c@subsubsection}
1545 \renewcommand{\theparagraph}{%
1546   \thesubsubsection.\@arabic\c@paragraph}

```

```
1547 \renewcommand{\thesubparagraph}{%
1548   \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。

```
1549 (book | report)\newcommand{\@chapapp}{\prechaptername}
```

```
1550 (book | report)\newcommand{\@chappos}{\postchaptername}
```

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

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

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

```
1551 (*book)
1552 \newcommand\frontmatter{%
1553   \pltx@cleartooddpage
1554   \@mainmatterfalse
1555   \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。

```
1556 \newcommand\mainmatter{%
1557   \pltx@cleartooddpage
1558   \@mainmattertrue
1559   \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。

```
1560 \newcommand\backmatter{%
1561   \if@openleft
1562     \cleardoublepage
1563   \else\if@openright
1564     \cleardoublepage
1565   \else
1566     \clearpage
1567   \fi\fi
1568   \@mainmatterfalse}
1569 </book>
```

■部

`\part` 新しい部を始めます。

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

`\secdef{星なし}{星あり}`

星なし * のない形の定義です。

星あり * のある形の定義です。

`\secdef` は次のようにして使います。

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

まず `book` と `report` のクラス以外です。

```
1570 (*!book&!report)
1571 \newcommand\part{%
1572   \if@noskipsec \leavevmode \fi
1573   \par
1574   \addvspace{4ex}%
1575   \if@english \@afterindentfalse \else \@afterindenttrue \fi
1576   \secdef\@part\@spart}
1577 </!book&!report)
```

`book` および `report` クラスの場合は、少し複雑です。

```
1578 (*book | report)
1579 \newcommand\part{%
1580   \if@openleft
1581     \cleardoublepage
1582   \else\if@openright
1583     \cleardoublepage
1584   \else
1585     \clearpage
1586   \fi\fi
1587   \thispagestyle{empty}% 欧文用標準スタイルでは plain
1588   \if@twocolumn
1589     \onecolumn
1590     \@restonecoltrue
1591   \else
1592     \@restonecolfalse
1593   \fi
1594   \null\vfil
1595   \secdef\@part\@spart}
1596 </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。

```
1597 (*!book&!report)
1598 \def\@part[#1]#2{%
1599   \ifnum \c@secnumdepth >\m@ne
1600     \refstepcounter{part}%
```

```

1601 \addcontentsline{toc}{part}{%
1602 \prepartname\thepart\postpartname\hspace{1zw}#1}%
1603 \else
1604 \addcontentsline{toc}{part}{#1}%
1605 \fi
1606 \markboth{}{}%
1607 {\parindent\z@
1608 \raggedright
1609 \interlinepenalty \@M
1610 \normalfont
1611 \ifnum \c@secnumdepth >\m@ne
1612 \Large\headfont\prepartname\thepart\postpartname
1613 \par\nobreak
1614 \fi
1615 \huge \headfont #2%
1616 \markboth{}{}\par}%
1617 \nobreak
1618 \vskip 3ex
1619 \@afterheading}
1620 \!/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。

```

1621 (*book | report)
1622 \def\@part[#1]#2{%
1623 \ifnum \c@secnumdepth >-2\relax
1624 \refstepcounter{part}%
1625 \addcontentsline{toc}{part}{%
1626 \prepartname\thepart\postpartname\hspace{1zw}#1}%
1627 \else
1628 \addcontentsline{toc}{part}{#1}%
1629 \fi
1630 \markboth{}{}%
1631 {\centering
1632 \interlinepenalty \@M
1633 \normalfont
1634 \ifnum \c@secnumdepth >-2\relax
1635 \huge\headfont \prepartname\thepart\postpartname
1636 \par\vskip20\jsc@empt
1637 \fi
1638 \Huge \headfont #2\par}%
1639 \@endpart}
1640 \!/book | report)

```

\@spart 番号を付けない部です。

```

1641 (*!book&!report)
1642 \def\@spart#1{%
1643 \parindent \z@ \raggedright
1644 \interlinepenalty \@M
1645 \normalfont

```

```

1646 \huge \headfont #1\par}%
1647 \nobreak
1648 \vskip 3ex
1649 \@afterheading}
1650 </!book&!report)
1651 (*book | report)
1652 \def\@spart#1{%
1653 \centering
1654 \interlinepenalty \@M
1655 \normalfont
1656 \Huge \headfont #1\par}%
1657 \@endpart}
1658 </book | report)

```

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

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

```

1659 (*book | report)
1660 \def\@endpart{\vfil\newpage
1661 \if@twoside
1662 \if@openleft %% added (2017/02/24)
1663 \null\thispagestyle{empty}\newpage
1664 \else\if@openright %% added (2016/12/13)
1665 \null\thispagestyle{empty}\newpage
1666 \fi\fi %% added (2016/12/13, 2017/02/24)
1667 \fi
1668 \if@restonecol
1669 \twocolumn
1670 \fi}
1671 </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。

```

1672 (*book | report)
1673 \newcommand{\chapter}{%
1674 \if@openleft\cleardoublepage\else
1675 \if@openright\cleardoublepage\else\clearpage\fi\fi
1676 \plainifnotempty % 元: \thispagestyle{plain}
1677 \global\@topnum\z@
1678 \if@english \@afterindentfalse \else \@afterindenttrue \fi
1679 \secdef
1680 {\@omit@numberfalse\@chapter}%
1681 {\@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。

```
1682 \def\@chapter[#1]#2{%
1683   \ifnum \c@secnumdepth >\m@ne
1684 (book)   \if@mainmatter
1685           \refstepcounter{chapter}%
1686           \typeout{\@chapapp\thechapter\@chappos}%
1687           \addcontentsline{toc}{chapter}%
1688             {\protect\numberline
1689              % {\if@english\thechapter\else\@chapapp\thechapter\@chappos\fi}%
1690              {\@chapapp\thechapter\@chappos}%
1691              #1}%
1692 (book)   \else\addcontentsline{toc}{chapter}{#1}\fi
1693   \else
1694     \addcontentsline{toc}{chapter}{#1}%
1695   \fi
1696   \chaptermark{#1}%
1697   \addtocontents{lof}{\protect\addvspace{10\jsc@empt}}%
1698   \addtocontents{lot}{\protect\addvspace{10\jsc@empt}}%
1699   \if@twocolumn
1700     \@topnewpage[\@makechapterhead{#2}]%
1701   \else
1702     \@makechapterhead{#2}%
1703     \@afterheading
1704   \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。

```
1705 \def\@makechapterhead#1{%
1706   \vspace*{2\Cvs}% 欧文は 50pt
1707   {\parindent \z@ \raggedright \normalfont
1708     \ifnum \c@secnumdepth >\m@ne
1709 (book)   \if@mainmatter
1710           \huge\headfont \@chapapp\thechapter\@chappos
1711           \par\nobreak
1712           \vskip \Cvs % 欧文は 20pt
1713 (book)   \fi
1714   \fi
1715   \interlinepenalty\M
1716   \Huge \headfont #1\par\nobreak
1717   \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。

```
1718 \def\@schapter#1{%
1719   \chaptermark{#1}%
1720   \if@twocolumn
1721     \@topnewpage[\@makeschapterhead{#1}]%
1722   \else
1723     \@makeschapterhead{#1}\@afterheading
1724   \fi}
```

`\@makeschapterhead` 番号なしの章見出しです。

```
1725 \def\@makeschapterhead#1{%
1726   \vspace*{2\Cvs}% 欧文は 50pt
1727   {\parindent \z@ \raggedright
1728     \normalfont
1729     \interlinepenalty\M
1730     \Huge \headfont #1\par\nobreak
1731     \vskip 3\Cvs}} % 欧文は 40pt
1732 </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。

```
1733 \if@twocolumn
1734   \newcommand{\section}{%
1735     (jpf)\ifx\maketitle\relax\else\maketitle\fi
1736     \@startsection{section}{1}{\z@}%
1737     (!kiyou)   {0.6\Cvs}{0.4\Cvs}%
1738     (kiyou)   {\Cvs}{0.5\Cvs}%
1739     %   {\normalfont\large\headfont\@secapp}}
1740     {\normalfont\large\headfont\raggedright}}
1741 \else
1742   \newcommand{\section}{%
1743     \if@slide\clearpage\fi
1744     \@startsection{section}{1}{\z@}%
1745     {\Cvs \@plus.5\Cdp \@minus.2\Cdp}% 前アキ
1746     {.5\Cvs \@plus.3\Cdp}% 後アキ
1747     %   {\normalfont\Large\headfont\@secapp}}
1748     {\normalfont\Large\headfont\raggedright}}
1749 \fi
```

`\subsection` 同上です。

```
1750 \if@twocolumn
1751   \newcommand{\subsection}{\@startsection{subsection}{2}{\z@}%
1752     {\z@}{\if@slide .4\Cvs \else \z@ \fi}%
1753     {\normalfont\normalsize\headfont}}
1754 \else
1755   \newcommand{\subsection}{\@startsection{subsection}{2}{\z@}%
1756     {\Cvs \@plus.5\Cdp \@minus.2\Cdp}% 前アキ
1757     {.5\Cvs \@plus.3\Cdp}% 後アキ
1758     {\normalfont\large\headfont}}
1759 \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)。

```
1760 \if@twocolumn
```

```

1761 \newcommand{\subsubsection}{\@startsection{subsubsection}{3}{\z@}%
1762   {\z@}{\if@slide .4\Cvs \else \z@ \fi}%
1763   {\normalfont\normalsize\headfont}}
1764 \else
1765 \newcommand{\subsubsection}{\@startsection{subsubsection}{3}{\z@}%
1766   {\Cvs \@plus.5\Cdp \@minus.2\Cdp}%
1767   {\if@slide .5\Cvs \@plus.3\Cdp \else \z@ \fi}%
1768   {\normalfont\normalsize\headfont}}
1769 \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。

```

1770 (!jspf)\newcommand{\jsParagraphMark}{■}
1771 \if@twocolumn
1772 \newcommand{\paragraph}{\@startsection{paragraph}{4}{\z@}%
1773   {\z@}{\if@slide .4\Cvs \else -1zw\fi}% 改行せず 1zw のアキ
1774 (jspf)   {\normalfont\normalsize\headfont}}
1775 (!jspf)   {\normalfont\normalsize\headfont\jsParagraphMark}}
1776 \else
1777 \newcommand{\paragraph}{\@startsection{paragraph}{4}{\z@}%
1778   {0.5\Cvs \@plus.5\Cdp \@minus.2\Cdp}%
1779   {\if@slide .5\Cvs \@plus.3\Cdp \else -1zw\fi}% 改行せず 1zw のアキ
1780 (jspf)   {\normalfont\normalsize\headfont}}
1781 (!jspf)   {\normalfont\normalsize\headfont\jsParagraphMark}}
1782 \fi

```

`\subparagraph` 見出しの後ろで改行されません。

```

1783 \if@twocolumn
1784 \newcommand{\subparagraph}{\@startsection{subparagraph}{5}{\z@}%
1785   {\z@}{\if@slide .4\Cvs \@plus.3\Cdp \else -1zw\fi}%
1786   {\normalfont\normalsize\headfont}}
1787 \else
1788 \newcommand{\subparagraph}{\@startsection{subparagraph}{5}{\z@}%
1789   {\z@}{\if@slide .5\Cvs \@plus.3\Cdp \else -1zw\fi}%
1790   {\normalfont\normalsize\headfont}}
1791 \fi

```

8.3 リスト環境

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

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

[2002-05-11] 3zw に変更しました。

[2005-03-19] 二段組は 2zw に戻しました。

```
1792 \if@slide
1793   \setlength\leftmargini{1zw}
1794 \else
1795   \if@twocolumn
1796     \setlength\leftmargini{2zw}
1797   \else
1798     \setlength\leftmargini{3zw}
1799   \fi
1800 \fi
```

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

```
\leftmarginiv 1801 \if@slide
\leftmarginv 1802   \setlength\leftmarginii {1zw}
1803   \setlength\leftmarginiii{1zw}
\leftmarginvi 1804   \setlength\leftmarginiv {1zw}
1805   \setlength\leftmarginv  {1zw}
1806   \setlength\leftmarginvi {1zw}
1807 \else
1808   \setlength\leftmarginii {2zw}
1809   \setlength\leftmarginiii{2zw}
1810   \setlength\leftmarginiv {2zw}
1811   \setlength\leftmarginv  {1zw}
1812   \setlength\leftmarginvi {1zw}
1813 \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。

```
1814 \setlength \labelsep {0.5zw} % .5em
1815 \setlength \labelwidth{\leftmargini}
1816 \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。

```
1817 \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 1818 \@beginparpenalty -\@lowpenalty
\@itempenalty 1819 \@endparpenalty -\@lowpenalty
1820 \@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。

```
1821 \def\@listi{\leftmargin\leftmargini
1822   \parsep \z@
1823   \topsep 0.5\baselineskip
1824   \itemsep \z@ \relax}
1825 \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)。

```
1826 \@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 1827 \def\@listii{\leftmargin\leftmarginii
\@listiv 1828   \labelwidth\leftmarginii \advance\labelwidth-\labelsep
1829   \topsep \z@
\@listv 1830   \parsep \z@
\@listvi 1831   \itemsep\parsep}
1832 \def\@listiii{\leftmargin\leftmarginiii
1833   \labelwidth\leftmarginiii \advance\labelwidth-\labelsep
1834   \topsep \z@
1835   \parsep \z@
1836   \itemsep\parsep}
1837 \def\@listiv {\leftmargin\leftmarginiv
1838   \labelwidth\leftmarginiv
1839   \advance\labelwidth-\labelsep}
1840 \def\@listv {\leftmargin\leftmarginv
1841   \labelwidth\leftmarginv
1842   \advance\labelwidth-\labelsep}
1843 \def\@listvi {\leftmargin\leftmarginvi
1844   \labelwidth\leftmarginvi
1845   \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。

```
1846 \renewcommand{\theenumi}{\@arabic\c@enumi}
1847 \renewcommand{\theenumii}{\@alph\c@enumii}
1848 \renewcommand{\theenumiii}{\@roman\c@enumiii}
1849 \renewcommand{\theenumiv}{\@Alph\c@enumiv}
```

`\labelenumi` `enumerate` 環境の番号を出力する命令です。第 2 レベル以外は最後に欧文のピリオドが付きますが、これは好みに応じて取り払ってください。第 2 レベルの番号のかっこは和文用に
`\labelenumii` 換え、その両側に入る余分なグルーを `\inhibitglue` で取り除いています。

```
\labelenumiv 1850 \newcommand{\labelenumi}{\theenumi.}
1851 \newcommand{\labelenumii}{\inhibitglue (\theenumii) \inhibitglue}
1852 \newcommand{\labelenumiii}{\theenumiii.}
1853 \newcommand{\labelenumiv}{\theenumiv.}
```

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

```
\p@enumiv 1854 \renewcommand{\p@enumii}{\theenumi}
1855 \renewcommand{\p@enumiii}{\theenumi\inhibitglue (\theenumii) }
1856 \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 1857 \newcommand\labelitemi{\textbullet}
\labelitemiii 1858 \newcommand\labelitemii{\normalfont\bfseries \textendash}
1859 \newcommand\labelitemiii{\textasteriskcentered}
\labelitemiv 1860 \newcommand\labelitemiv{\textperiodcentered}
```

■description 環境

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

```
1861 \newenvironment{description}{%
1862 \list{}{%
1863 \labelwidth=\leftmargin
1864 \labelsep=1zw
1865 \advance \labelwidth by -\labelsep
1866 \let \makelabel=\descriptionlabel}}{\endlist}
```

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

```
1867 \newcommand*\descriptionlabel[1]{\normalfont\headfont #1\hfil}
```

■概要

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

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

```
1868 (*book)
1869 \newenvironment{abstract}{%
```

```

1870 \begin{list}{-}{%
1871   \listparindent=1zw
1872   \itemindent=\listparindent
1873   \rightmargin=0pt
1874   \leftmargin=5zw}\item[]}{\end{list}\vspace{\baselineskip}}
1875 
```

1876 (*article | report | kiyou)
1877 \newbox\@abstractbox
1878 \if@titlepage
1879 \newenvironment{abstract}{%
1880 \titlepage
1881 \null\vfil
1882 \@beginparpenalty\@lowpenalty
1883 \begin{center}%
1884 \headfont \abstractname
1885 \@endparpenalty\@M
1886 \end{center}}%
1887 {\par\vfil\null\endtitlepage}
1888 \else
1889 \newenvironment{abstract}{%
1890 \if@twocolumn
1891 \ifx\maketitle\relax
1892 \section*{\abstractname}%
1893 \else
1894 \global\setbox\@abstractbox\hbox\bgroup
1895 \begin{minipage}[b]{\textwidth}
1896 \small\parindent1zw
1897 \begin{center}%
1898 {\headfont \abstractname\vspace{-.5em}\vspace{\z@}}%
1899 \end{center}%
1900 \list{}{%
1901 \listparindent\parindent
1902 \itemindent \listparindent
1903 \rightmargin \leftmargin}%
1904 \item\relax
1905 \fi
1906 \else
1907 \small
1908 \begin{center}%
1909 {\headfont \abstractname\vspace{-.5em}\vspace{\z@}}%
1910 \end{center}%
1911 \list{}{%
1912 \listparindent\parindent
1913 \itemindent \listparindent
1914 \rightmargin \leftmargin}%
1915 \item\relax
1916 \fi}{\if@twocolumn
1917 \ifx\maketitle\relax
1918 \else

```

1919         \endlist\end{minipage}\egroup
1920     \fi
1921     \else
1922     \endlist
1923     \fi}
1924 \fi
1925 </article | report | kiyou>
1926 <*jspf>
1927 \newbox\@abstractbox
1928 \newenvironment{abstract}{%
1929     \global\setbox\@abstractbox\hbox\bgroup
1930     \begin{minipage}[b]{157\jsc@mmm}{\sffamily Abstract}\par
1931     \small
1932     \if@english \parindent6\jsc@mmm \else \parindent1zw \fi}%
1933     {\end{minipage}\egroup}
1934 </jspf>

```

■キーワード

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

```

1935 <*jspf>
1936 %\newbox\@keywordsbox
1937 %\newenvironment{keywords}{%
1938 %     \global\setbox\@keywordsbox\hbox\bgroup
1939 %     \begin{minipage}[b]{1570\jsc@mmm}{\sffamily Keywords:}\par
1940 %     \small\parindent0zw}%
1941 %     {\end{minipage}\egroup}
1942 </jspf>

```

■verse 環境

verse 詩のための `verse` 環境です。

```

1943 \newenvironment{verse}{%
1944     \let \\\=@centercr
1945     \list{}{%
1946         \itemsep \z@
1947         \itemindent -2zw % 元: -1.5em
1948         \listparindent\itemindent
1949         \rightmargin \z@
1950         \advance\leftmargin 2zw}% 元: 1.5em
1951     \item\relax}{\endlist}

```

■quotation 環境

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

```

1952 \newenvironment{quotation}{%
1953     \list{}{%

```

```

1954 \listparindent\parindent
1955 \itemindent\listparindent
1956 \rightmargin \z0}%
1957 \item\relax}{\endlist}

```

■quote 環境

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

```

1958 \newenvironment{quote}%
1959 {\list{}{\rightmargin\z0}\item\relax}{\endlist}

```

■定理など ltthm.dtx 参照。たとえば次のように定義します。

```

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

```

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

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

```

1960 \def\@begintheorem#1#2{\trivlist\labelsep=1zw
1961 \item[\hskip \labelsep{\headfont #1\ #2}]}
1962 \def\@opargbegintheorem#1#2#3{\trivlist\labelsep=1zw
1963 \item[\hskip \labelsep{\headfont #1\ #2 (#3)}]}

```

titlepage タイトルを独立のページに出力するのに使われます。

[2017-02-24] コミュニティ版 pL^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、こちらでも片面印刷あるいは独立のタイトルページを作らないクラスばかりでしたので、従来の挙動は何も変わらずに済みました。

```

1964 \newenvironment{titlepage}{%
1965 (book) \pltx@cleartooddpage %% 2017-02-24
1966 \if@twocolumn
1967 \@restonecoltrue\onecolumn
1968 \else
1969 \@restonecolfalse\newpage
1970 \fi
1971 \thispagestyle{empty}%
1972 \ifodd\c@page\setcounter{page}\@ne\else\setcounter{page}\z@\fi %% 2017-02-24
1973 }%
1974 {\if@restonecol\twocolumn \else \newpage \fi
1975 \if@twoside\else
1976 \setcounter{page}\@ne
1977 \fi}

```

■付録

`\appendix` 本文と付録を分離するコマンドです。

```
1978 <!*book&!report)
1979 \newcommand{\appendix}{\par
1980   \setcounter{section}{0}%
1981   \setcounter{subsection}{0}%
1982   \gdef\presectionname{\appendixname}%
1983   \gdef\postsectionname{}}%
1984 % \gdef\thesection{\@Alph\c@section}% [2003-03-02]
1985   \gdef\thesection{\presectionname\@Alph\c@section\postsectionname}%
1986   \gdef\thesubsection{\@Alph\c@section.\@arabic\c@subsection}}
1987 </!book&!report)
1988 <*book | report)
1989 \newcommand{\appendix}{\par
1990   \setcounter{chapter}{0}%
1991   \setcounter{section}{0}%
1992   \gdef\@chapapp{\appendixname}%
1993   \gdef\@chappos{}}%
1994   \gdef\thechapter{\@Alph\c@chapter}}
1995 </book | report)
```

8.4 パラメータの設定

■array と tabular 環境

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

```
1996 \setlength\arraycolsep{5\jsc@mp}
```

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

```
1997 \setlength\tabcolsep{6\jsc@mp}
```

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

```
1998 \setlength\arrayrulewidth{.4\jsc@mp}
```

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

```
1999 \setlength\doublerulesep{2\jsc@mp}
```

■tabbing 環境

`\tabbingsep` \’ コマンドで入るアキです。

```
2000 \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。

```
2001 \skip\@mpfootins = \skip\footins
```

■framebox 環境

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

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

```
2002 \setlength\fboxsep{3\jsc@empt}
2003 \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。

```
2004 (!book&!report)\renewcommand \theequation {\@arabic\c@equation}
2005 (*book | report)
2006 \@addtoreset{equation}{chapter}
2007 \renewcommand\theequation
2008 {\ifnum \c@chapter>\z@ \thechapter.\fi \@arabic\c@equation}
2009 </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。

```
2010 % \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。

```
2011 % \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。

```
2012 % \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。

```
2013 <!*book&!report>
2014 \newcounter{figure}
2015 \renewcommand \thefigure {\@arabic\c@figure}
2016 </!book&!report>
2017 <*book | report>
2018 \newcounter{figure}[chapter]
2019 \renewcommand \thefigure
2020     {\ifnum \c@chapter>\z@ \thechapter.\fi \@arabic\c@figure}
2021 </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。
`\ftype@figure`

```
\ext@figure 2022 \def\fps@figure{tbp}
2023 \def\ftype@figure{1}
\fnun@figure 2024 \def\ext@figure{lof}
2025 \def\fnun@figure{\figurename\nobreak\thefigure}
```

`figure` * 形式は段抜きフロートです。

```
figure* 2026 \newenvironment{figure}%
2027     {\@float{figure}}%
2028     {\end@float}
2029 \newenvironment{figure*}%
2030     {\@dblfloat{figure}}%
2031     {\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。

```
2032 <!*book&!report>
2033 \newcounter{table}
2034 \renewcommand\thetable{\@arabic\c@table}
2035 </!book&!report>
2036 <*book | report>
2037 \newcounter{table}[chapter]
2038 \renewcommand \thetable
2039     {\ifnum \c@chapter>\z@ \thechapter.\fi \@arabic\c@table}
2040 </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`

```
\ext@table 2041 \def\fps@table{tbp}
2042 \def\ftype@table{2}
\fnun@table 2043 \def\ext@table{lot}
2044 \def\fnun@table{\tablename\nobreak\thetable}
```

`table` * は段抜きフロートです。

`table*`

```

2045 \newenvironment{table}%
2046         {\@float{table}}%
2047         {\end@float}
2048 \newenvironment{table*}%
2049         {\@dblfloat{table}}%
2050         {\end@dblfloat}

```

8.6 キャプション

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

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

```

2051 \newlength\abovecaptionskip
2052 \newlength\belowcaptionskip
2053 \setlength\abovecaptionskip{5\jsc@empt} % 元: 10\p@
2054 \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。

```

2055 (*!jspf)
2056 % \long\def\@makecaption#1#2{\small
2057 %   \advance\leftskip10\jsc@mmm
2058 %   \advance\rightskip10\jsc@mmm
2059 %   \vskip\abovecaptionskip
2060 %   \sbox\@tempboxa{#1\hskip1zw\relax #2}%
2061 %   \ifdim \wd\@tempboxa >\hsiz
2062 %     #1\hskip1zw\relax #2\par
2063 %   \else
2064 %     \global \@minipagefalse
2065 %     \hb@xt@\hsiz{\hfil\box\@tempboxa\hfil}%
2066 %   \fi
2067 %   \vskip\belowcaptionskip}}
2068 \long\def\@makecaption#1#2{\small
2069   \advance\leftskip .0628\linewidth
2070   \advance\rightskip .0628\linewidth
2071   \vskip\abovecaptionskip
2072   \sbox\@tempboxa{#1\hskip1zw\relax #2}%
2073   \ifdim \wd\@tempboxa <\hsiz \centering \fi
2074   #1{\hskip1zw\relax}#2\par
2075   \vskip\belowcaptionskip}}

```

```

2076 </jspf>
2077 <*jspf>
2078 \long\def\@makecaption#1#2{%
2079   \vskip\abovecaptionskip
2080   \sbox\@tempboxa{\small\sffamily #1\quad #2}%
2081   \ifdim \wd\@tempboxa >\hsize
2082     {\small\sffamily
2083       \list{#1}{%
2084         \renewcommand{\makelabel}[1]{##1\hfil}
2085         \itemsep \z@
2086         \itemindent \z@
2087         \labelsep \z@
2088         \labelwidth 11\jsc@mmm
2089         \listparindent\z@
2090         \leftmargin 11\jsc@mmm}\item\relax #2\endlist}
2091   \else
2092     \global \@minipagefalse
2093     \hb@xt@\hsize{\hfil\box\@tempboxa\hfil}%
2094   \fi
2095   \vskip\belowcaptionskip}
2096 </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 2097 \DeclareOldFontCommand{\mc}{\normalfont\mcfamily}{\mathmc}
\rm 2098 \DeclareOldFontCommand{\gt}{\normalfont\gtfamily}{\mathgt}
\sf 2099 \DeclareOldFontCommand{\rm}{\normalfont\rmfamily}{\mathrm}
\sf 2100 \DeclareOldFontCommand{\sf}{\normalfont\sffamily}{\mathsf}
\tt 2101 \DeclareOldFontCommand{\tt}{\normalfont\ttfamily}{\mathtt}

```

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

```

2102 \DeclareOldFontCommand{\bf}{\normalfont\bfseries}{\mathbf}

```

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

```

2103 \DeclareOldFontCommand{\it}{\normalfont\itshape}{\mathit}
2104 \DeclareOldFontCommand{\sl}{\normalfont\slshape}{\@nomath\sl}
2105 \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`

```
2106 \DeclareRobustCommand*\cal{\@fontswitch\relax\mathcal}
2107 \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。

```
2108 \newcommand\@pnumwidth{1.55em}
2109 \newcommand\@tocrmarg{2.55em}
2110 \newcommand\@dotsep{4.5}
2111 (!book&!report)\setcounter{tocdepth}{2}
2112 (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)

```
2113 \newdimen\jsc@tocl@width
2114 \newcommand{\tableofcontents}{%
2115 (*book | report)
2116   \settowidth\jsc@tocl@width{\headfont\prechaptername\postchaptername}%
2117   \settowidth\@tempdima{\headfont\appendixname}%
2118   \ifdim\jsc@tocl@width<\@tempdima \setlength\jsc@tocl@width{\@tempdima}\fi
2119   \ifdim\jsc@tocl@width<2zw \divide\jsc@tocl@width by 2 \advance\jsc@tocl@width 1zw\fi
2120   \if@twocolumn
2121     \@restonecoltrue\onecolumn
2122   \else
2123     \@restonecolfalse
2124   \fi
2125   \chapter*{\contentsname}%
2126   \@mkboth{\contentsname}{}%
2127 </book | report)
2128 (*!book&!report)
2129   \settowidth\jsc@tocl@width{\headfont\presectionname\postsectionname}%
2130   \settowidth\@tempdima{\headfont\appendixname}%
2131   \ifdim\jsc@tocl@width<\@tempdima\relax\setlength\jsc@tocl@width{\@tempdima}\fi
2132   \ifdim\jsc@tocl@width<2zw \divide\jsc@tocl@width by 2 \advance\jsc@tocl@width 1zw\fi
2133   \section*{\contentsname}%
2134   \@mkboth{\contentsname}{\contentsname}%
2135 </!book&!report)
2136   \@starttoc{toc}%
2137 (book | report) \if@restonecol\twocolumn\fi
2138 }
```

`\l@part` 部の目次です。

```
2139 \newcommand*{\l@part}[2]{%
2140   \ifnum \c@tocdepth >-2\relax
2141     (!book&!report) \addpenalty\@secpenalty
2142     (book | report) \addpenalty{-\@highpenalty}%
2143     \addvspace{2.25em \@plus\jsc@mpt}%
2144     \begingroup
2145       \parindent \z@
2146     % \@pnumwidth should be \@tocrmarg
2147     % \rightskip \@pnumwidth
2148     \rightskip \@tocrmarg
2149     \parfillskip -\rightskip
2150     {\leavevmode
2151       \large \headfont
2152       \setlength\@lnumwidth{4zw}%
2153       #1\hfil \hb@xt@\@pnumwidth{\hss #2}}\par
2154     \nobreak
```

```

2155 <book | report> \global\@nobreaktrue
2156 <book | report> \everypar{\global\@nobreakfalse\everypar{}}%
2157 \endgroup
2158 \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)

```

2159 <*book | report>
2160 \newcommand*{\l@chapter}[2]{%
2161 \ifnum \c@tocdepth >\m@ne
2162 \addpenalty{-\@highpenalty}%
2163 \addvspace{1.0em \@plus\jsc@empt}
2164 % \vskip 1.0em \@plus\p@ % book.cls では↑がこうなっている
2165 \begingroup
2166 \parindent\z@
2167 % \rightskip\@pnumwidth
2168 \rightskip\@tocrmarg
2169 \parfillskip-\rightskip
2170 \leavevmode\headfont
2171 % \if@english\setlength\@lnumwidth{5.5em}\else\setlength\@lnumwidth{4.683zw}\fi
2172 \setlength\@lnumwidth{\jsc@tocl@width}\advance\@lnumwidth 2.683zw
2173 \advance\leftskip\@lnumwidth \hskip-\leftskip
2174 #1\nobreak\hfil\nobreak\hbox to\@pnumwidth{\hss#2}\par
2175 \penalty\@highpenalty
2176 \endgroup
2177 \fi}
2178 </book | report>

```

`\l@section` 節の目次です。

```

2179 <!*book&!report>
2180 \newcommand*{\l@section}[2]{%
2181 \ifnum \c@tocdepth >\z@
2182 \addpenalty{\@secpenalty}%
2183 \addvspace{1.0em \@plus\jsc@empt}%
2184 \begingroup
2185 \parindent\z@
2186 % \rightskip\@pnumwidth
2187 \rightskip\@tocrmarg
2188 \parfillskip-\rightskip
2189 \leavevmode\headfont
2190 %\setlength\@lnumwidth{4zw}% 元 1.5em [2003-03-02]
2191 \setlength\@lnumwidth{\jsc@tocl@width}\advance\@lnumwidth 2zw
2192 \advance\leftskip\@lnumwidth \hskip-\leftskip
2193 #1\nobreak\hfil\nobreak\hbox to\@pnumwidth{\hss#2}\par
2194 \endgroup
2195 \fi}
2196 </!*book&!report>

```

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

2197 (book | report) % \newcommand*{\l@section}{\@dottedtocline{1}{1zw}{3.683zw}}

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

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

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

\l@subparagraph 2198 (*!book&!report)

2199 % \newcommand*{\l@subsection} {\@dottedtocline{2}{1.5em}{2.3em}}

2200 % \newcommand*{\l@subsubsection}{\@dottedtocline{3}{3.8em}{3.2em}}

2201 % \newcommand*{\l@paragraph} {\@dottedtocline{4}{7.0em}{4.1em}}

2202 % \newcommand*{\l@subparagraph} {\@dottedtocline{5}{10em}{5em}}

2203 %

2204 % \newcommand*{\l@subsection} {\@dottedtocline{2}{1zw}{3zw}}

2205 % \newcommand*{\l@subsubsection}{\@dottedtocline{3}{2zw}{3zw}}

2206 % \newcommand*{\l@paragraph} {\@dottedtocline{4}{3zw}{3zw}}

2207 % \newcommand*{\l@subparagraph} {\@dottedtocline{5}{4zw}{3zw}}

2208 %

2209 \newcommand*{\l@subsection}{%

2210 \@tempdima\jsc@tocl@width \advance\@tempdima -1zw

2211 \@dottedtocline{2}{\@tempdima}{3zw}}

2212 \newcommand*{\l@subsubsection}{%

2213 \@tempdima\jsc@tocl@width \advance\@tempdima 0zw

2214 \@dottedtocline{3}{\@tempdima}{4zw}}

2215 \newcommand*{\l@paragraph}{%

2216 \@tempdima\jsc@tocl@width \advance\@tempdima 1zw

2217 \@dottedtocline{4}{\@tempdima}{5zw}}

2218 \newcommand*{\l@subparagraph}{%

2219 \@tempdima\jsc@tocl@width \advance\@tempdima 2zw

2220 \@dottedtocline{5}{\@tempdima}{6zw}}

2221 \!/book&!report)

2222 (*book | report)

2223 % \newcommand*{\l@subsection} {\@dottedtocline{2}{3.8em}{3.2em}}

2224 % \newcommand*{\l@subsubsection}{\@dottedtocline{3}{7.0em}{4.1em}}

2225 % \newcommand*{\l@paragraph} {\@dottedtocline{4}{10em}{5em}}

2226 % \newcommand*{\l@subparagraph} {\@dottedtocline{5}{12em}{6em}}

2227 \newcommand*{\l@section}{%

2228 \@tempdima\jsc@tocl@width \advance\@tempdima -1zw

2229 \@dottedtocline{1}{\@tempdima}{3.683zw}}

2230 \newcommand*{\l@subsection}{%

2231 \@tempdima\jsc@tocl@width \advance\@tempdima 2.683zw

2232 \@dottedtocline{2}{\@tempdima}{3.5zw}}

2233 \newcommand*{\l@subsubsection}{%

2234 \@tempdima\jsc@tocl@width \advance\@tempdima 6.183zw

2235 \@dottedtocline{3}{\@tempdima}{4.5zw}}

2236 \newcommand*{\l@paragraph}{%

2237 \@tempdima\jsc@tocl@width \advance\@tempdima 10.683zw

2238 \@dottedtocline{4}{\@tempdima}{5.5zw}}

```

2239 \newcommand*\l@subparagraph}{%
2240     \@tempdima\jsc@tocl@width \advance\@tempdima 16.183zw
2241     \@dottedtocline{5}{\@tempdima}{6.5zw}}
2242 (/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。

```

2243 \newdimen\@lnumwidth
2244 \def\numberline#1{\hb@xt@\@lnumwidth{#1\hfil}\hspace{0pt}}

```

`\@dottedtocline` L^AT_EX 本体 (l^tsect.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、仮想ポディの中央 に変更したい場合は

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

とします。

```

2245 \def\jsTocLine{\leaders\hbox{%
2246     $\m@th \mkern \@dotsep mu\hbox{.}\mkern \@dotsep mu$}\hfill}
2247 \def\@dottedtocline#1#2#3#4#5{\ifnum #1>\c@tocdepth \else
2248     \vskip \z@ \@plus.2\jsc@mpt
2249     {\leftskip #2\relax \rightskip \@tocrmarg \parfillskip -\rightskip
2250     \parindent #2\relax\@afterindenttrue
2251     \interlinepenalty\@M
2252     \leavevmode
2253     \@lnumwidth #3\relax
2254     \advance\leftskip \@lnumwidth \null\nobreak\hskip -\leftskip
2255     {#4}\nobreak
2256     \jsTocLine \nobreak\hb@xt@\@pnumwidth{%
2257     \hfil\normalfont \normalcolor #5}\par}\fi}

```

■ 図目次と表目次

`\listoffigures` 図目次を出力します。

```

2258 \newcommand{\listoffigures}{%
2259 (*book | report)
2260     \iftwocolumn\@restonecoltrue\onecolumn
2261     \else\@restonecolfalse\fi
2262     \chapter*{\listfigurename}%
2263     \@mkboth{\listfigurename}{}%
2264 (/book | report)
2265 (*!book&!report)
2266     \section*{\listfigurename}%
2267     \@mkboth{\listfigurename}{\listfigurename}%

```

```

2268 </!book&!report)
2269 \starttoc{lof}%
2270 (book | report) \if@restonecol\twocolumn\fi
2271 }

```

`\l@figure` 図目次の項目を出力します。

```
2272 \newcommand*\l@figure{\@dottedtocline{1}{1zw}{3.683zw}}
```

`\listoftables` 表目次を出力します。

```

2273 \newcommand\listoftables{%
2274 (*book | report)
2275 \if@twocolumn\@restonecoltrue\onecolumn
2276 \else\@restonecolfalse\fi
2277 \chapter*\listtablename}%
2278 \@mkboth{\listtablename}{}%
2279 </book | report)
2280 (*!book&!report)
2281 \section*\listtablename}%
2282 \@mkboth{\listtablename}{\listtablename}%
2283 </!book&!report)
2284 \starttoc{lot}%
2285 (book | report) \if@restonecol\twocolumn\fi
2286 }

```

`\l@table` 表目次は図目次と同じです。

```
2287 \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。

```

2288 \newdimen\bibindent
2289 \setlength\bibindent{2zw}

```

`thebibliography` 参考文献リストを出力します。

```

2290 \newenvironment{thebibliography}[1]{%
2291 \global\let\presectionname\relax
2292 \global\let\postsectionname\relax
2293 (article | jspf) \section*\refname\@mkboth{\refname}{\refname}%
2294 (*kiyou)
2295 \vspace{1.5\baselineskip}
2296 \subsubsection*\refname\@mkboth{\refname}{\refname}%
2297 \vspace{0.5\baselineskip}
2298 </kiyou)
2299 (book | report) \chapter*\bibname\@mkboth{\bibname}{}%
2300 (book | report) \addcontentsline{toc}{chapter}{\bibname}%
2301 \list{\@biblabel{\@arabic\c@enumiv}}%
2302 {\settowidth\labelwidth{\@biblabel{#1}}%
2303 \leftmargin\labelwidth

```

```

2304      \advance\leftmargin\labelsep
2305      \@openbib@code
2306      \usecounter{enumiv}%
2307      \let\p@enumiv\@empty
2308      \renewcommand\theenumiv{\@arabic\c@enumiv}%
2309 (kiyou) \small
2310      \sloppy
2311      \clubpenalty4000
2312      \@clubpenalty\clubpenalty
2313      \widowpenalty4000%
2314      \sfcode'\.\@m}
2315      {\def\@noitemerr
2316      {\@latex@warning{Empty 'thebibliography' environment}}}%
2317      \endlist}

```

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

```
2318 \newcommand{\newblock}{\hskip .11em\@plus.33em\@minus.07em}
```

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

```
2319 \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。

```
2320 % \def\@biblabel#1{\inhibitglue [#1] \inhibitglue}
```

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

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

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

```

2321 % \def\@citex[#1]#2{\leavevmode
2322 %      \let\@citea\@empty
2323 %      \@cite{\@for\@citeb:=#2\do
2324 %          {\@citea\def\@citea{, \inhibitglue\penalty\@m\ }%
2325 %              \edef\@citeb{\expandafter\@firstofone\@citeb\@empty}}%
2326 %          \if@filesw\immediate\write\@auxout{\string\citation{\@citeb}}\fi
2327 %          \@ifundefined{b@\@citeb}{\mbox{\normalfont\bfseries ?}}%
2328 %              \G@refundefinedtrue
2329 %              \@latex@warning
2330 %                  {Citation '\@citeb' on page \thepage \space undefined}}%
2331 %          {\@cite#1\ofmt{\csname b@\@citeb\endcsname}}}{#1}}
2332 % \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。

```

2333 % \DeclareRobustCommand\cite{\unskip
2334 %      \@ifnextchar [{\@tempswatrue\@citex}{\@tempswafalse\@citex[]}]

```

```

2335 % \def\@cite#1#2{${\hbox{\scriptsize{#1}\if@tempswa
2336 %   , \inhibitglue\ #2\fi} )}}$}

```

10.3 索引

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

```

2337 \newenvironment{theindex}{% 索引を 3 段組で出力する環境
2338   \if@twocolumn
2339     \onecolumn\@restonecolfalse
2340   \else
2341     \clearpage\@restonecoltrue
2342   \fi
2343   \columnseprule.4pt \columnsep 2zw
2344   \ifx\multicols\@undefined
2345 (book | report)   \twocolumn[\@makeschapterhead{\indexname}%
2346 (book | report)   \addcontentsline{toc}{chapter}{\indexname}]%
2347 (!book&!report)   \def\presectionname{}\def\postsectionname{}%
2348 (!book&!report)   \twocolumn[\section*{\indexname}]%
2349   \else
2350     \ifdim\textwidth<\fullwidth
2351       \setlength{\evensidemargin}{\oddsidemargin}
2352       \setlength{\textwidth}{\fullwidth}
2353       \setlength{\linewidth}{\fullwidth}
2354 (book | report)   \begin{multicols}{3}[\chapter*{\indexname}]%
2355 (book | report)   \addcontentsline{toc}{chapter}{\indexname}]%
2356 (!book&!report)   \def\presectionname{}\def\postsectionname{}%
2357 (!book&!report)   \begin{multicols}{3}[\section*{\indexname}]%
2358   \else
2359 (book | report)   \begin{multicols}{2}[\chapter*{\indexname}]%
2360 (book | report)   \addcontentsline{toc}{chapter}{\indexname}]%
2361 (!book&!report)   \def\presectionname{}\def\postsectionname{}%
2362 (!book&!report)   \begin{multicols}{2}[\section*{\indexname}]%
2363   \fi
2364   \fi
2365 (book | report)   \@mkboth{\indexname}{}%
2366 (!book&!report)   \@mkboth{\indexname}{\indexname}%
2367   \plainifnotempty % \thispagestyle{plain}
2368   \parindent\z@
2369   \parskip\z@ \@plus .3\jcs@mpt\relax
2370   \let\item\@idxitem
2371   \raggedright
2372   \footnotesize\narrowbaselines
2373 }{
2374   \ifx\multicols\@undefined
2375     \if@restonecol\onecolumn\fi
2376   \else

```

```

2377     \end{multicols}
2378     \fi
2379     \clearpage
2380 }

```

`\@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 2381 \newcommand{\@idxitem}{\par\hangindent 4zw} % 元 40pt
\subsubitem 2382 \newcommand{\subitem}{\@idxitem \hspace*{2zw}} % 元 20pt
2383 \newcommand{\subsubitem}{\@idxitem \hspace*{3zw}} % 元 30pt

```

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

```

2384 \newcommand{\indexspace}{\par \vskip 10\jsc@mpt \@plus5\jsc@mpt \@minus3\jsc@mpt\relax}

```

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

```

2385 \newcommand\seename{\if@english see\else → \fi}
2386 \newcommand\alsename{\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。

```

2387 \@ifl@t@r\pfmtversion{2016/09/03}
2388     {\jsc@needspace@tchfalse}{\jsc@needspace@tchtrue}
2389 \ifjsc@needspace@tch
2390     \let\footnotes@ve=\footnote
2391     \def\footnote{\inhibitglue\footnotes@ve}
2392     \let\footnotemarks@ve=\footnotemark
2393     \def\footnotemark{\inhibitglue\footnotemarks@ve}
2394 \fi

```

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

[2002-04-09] インプリメントの仕方を変えたため消しました。

[2013-04-23] 新しい pT_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。

```

2395 \@ifl@t@r\pfmtversion{2016/04/17}
2396     {\jsc@needspace@tchfalse}{\jsc@needspace@tchtrue}

```

```

2397 \ifjsc@needspace@tch
2398 \renewcommand\@makefnmark{%
2399   \ifydir \hbox{\hbox{\@textsuperscript{\normalfont\@thefnmark}}\hbox{}}%
2400   \else\hbox{\yoko\@textsuperscript{\normalfont\@thefnmark}}\fi}
2401 \fi

```

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

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

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

```

2402 \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。

```

2403 % \def\thefootnote{\ifnum\c@footnote>\z@ 注 \kern0.1zw\@arabic\c@footnote\fi}

```

`\footnoterule` 本文と脚注の間の罫線です。

```

2404 \renewcommand{\footnoterule}{%
2405   \kern-3\jsc@empt
2406   \hrule width .4\columnwidth height 0.4\jsc@empt
2407   \kern 2.6\jsc@empt}

```

`\c@footnote` 脚注番号は章ごとにリセットされます。

```

2408 (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, *TeX and TUG NEWS*, Vol. 2, No. 4 (1993), p. 9

[2016-08-25] コミュニティ版 p_{La}TeX 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_{La}TeX のバグ修正に追随しました。

[2016-11-29] 古い p_{La}TeX 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。

```

2409 \long\def\@footnotetext{%
2410   \insert\footins\bgroup
2411     \normalfont\footnotesize
2412     \interlinepenalty\interfootnotelinepenalty
2413     \splittopskip\footnotesep
2414     \splitmaxdepth \dp\strutbox \floatingpenalty \@MM
2415     \hsize\columnwidth \@parboxrestore
2416     \protected@edef\@currentlabel{%
2417       \csname p@footnote\endcsname\@thefnmark
2418     }%

```

```

2419 \color@begingroup
2420 \makefnfntext{%
2421 \rule{z@}\footnotesep\ignorespaces}%
2422 \futurelet\jsc@next\jsc@fo@t}
2423 \def\jsc@fo@t{\ifcat\bgroup\noexpand\jsc@next \let\jsc@next\jsc@f@t
2424 \else \let\jsc@next\jsc@f@t\fi \jsc@next}
2425 \def\jsc@f@t{\bgroup\aftergroup\jsc@@foot\let\jsc@next}
2426 \def\jsc@f@t#1{#1\jsc@@foot}
2427 \def\jsc@@foot{\@finalstrut\strutbox\color@endgroup\egroup
2428 \ifx\pltx@foot@penalty\undefined\else
2429 \ifhmode\null\fi
2430 \ifnum\pltx@foot@penalty=z@\else
2431 \penalty\pltx@foot@penalty
2432 \pltx@foot@penalty\z@
2433 \fi
2434 \fi}

```

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

```

2435 \newcommand\makefnfntext[1]{%
2436 \advance\leftskip 3zw
2437 \parindent 1zw
2438 \noindent
2439 \llap{\makefnmark\hskip0.3zw}#1}

```

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

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

[2002-04-09] インプリメントの仕方を変えたため消しました。

```

2440 % \def\xfootnotenext[#1]{%
2441 % \begingroup
2442 % \ifnum#1>z@
2443 % \csname c@\mpfn\endcsname #1\relax
2444 % \unrestored@protected@xdef\@thefnmark{\thempfn}%
2445 % \else
2446 % \unrestored@protected@xdef\@thefnmark{}%
2447 % \fi
2448 % \endgroup
2449 % \@footnotetext}

```

11 段落の頭へのグルー挿入禁止

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

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

のすべてに拡大しました。

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

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

```
2450 \def\@inhibitglue{%
2451   \futurelet\@let@token\@inhibitglue}
2452 \begingroup
2453 \let\GDEF=\gdef
2454 \let\CATCODE=\catcode
2455 \let\ENDGROUP=\endgroup
2456 \CATCODE'k=12
2457 \CATCODE'a=12
2458 \CATCODE'n=12
2459 \CATCODE'j=12
2460 \CATCODE'i=12
2461 \CATCODE'c=12
2462 \CATCODE'h=12
2463 \CATCODE'r=12
2464 \CATCODE't=12
2465 \CATCODE'e=12
2466 \GDEF\KANJI@CHARACTER{kanji character }
2467 \ENDGROUP
2468 \def\@@inhibitglue{%
2469   \expandafter\expandafter\expandafter\jsc@inhibitglue\expandafter\meaning\expandafter\@let@token
2470   \expandafter\def\expandafter\jsc@inhibitglue\expandafter#\expandafter1\KANJI@CHARACTER#2#3\jsc@ig@temp
2471   \def\jsc@ig@temp{#1}%
2472   \ifx\jsc@ig@temp@empty
2473     \ifnum\the\inhibitxscode'#2=2\relax
2474       \inhibitglue
2475     \fi
2476   \fi}
2477 \let\everyparhook=\@inhibitglue
2478 \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。

```
2479 \def\@doendpe{%
2480   \@endpetrue
2481   \def\par{%
2482     \@restorepar\clubpenalty\@clubpenalty\everypar{\everyparhook}\par\@endpefalse}%
2483   \everypar{\setbox\z@\lastbox}\everypar{\everyparhook}\@endpefalse\everyparhook}}
```

[2017-08-31] minipage 環境にも対策します。

```
2484 \def\@setminipage{%
2485   \@minipagetrue
2486   \everypar{\@minipagefalse\everypar{\everyparhook}}%
2487 }
```

\item 命令の直後です。

```
2488 \def\@item[#1]{%
2489   \if@noperitem
2490     \@donoperitem
2491   \else
2492     \if@inlabel
2493       \indent \par
2494     \fi
2495     \ifhmode
2496       \unskip\unskip \par
2497     \fi
2498     \if@newlist
2499       \if@nobreak
2500         \@nbitem
2501       \else
2502         \addpenalty\@beginparpenalty
2503         \addvspace\@topsep
2504         \addvspace{-\parskip}%
2505       \fi
2506     \else
2507       \addpenalty\@itempenalty
2508       \addvspace\itemsep
2509     \fi
2510     \global\@inlabeltrue
2511   \fi
2512   \everypar{%
2513     \@minipagefalse
2514     \global\@newlistfalse
2515     \if@inlabel
2516       \global\@inlabelfalse
2517       {\setbox\z@\lastbox
2518        \ifvoid\z@
2519          \kern-\itemindent
2520        \fi}%
2521     \box\@labels
2522     \penalty\z@
2523   \fi
2524   \if@nobreak
2525     \@nobreakfalse
2526     \clubpenalty \@M
2527   \else
2528     \clubpenalty \@clubpenalty
2529     \everypar{\everyparhook}%
```

```

2530 \fi\everyparhook}%
2531 \if@noitemarg
2532 \@noitemargfalse
2533 \if@nmbrrlist
2534 \refstepcounter\@listctr
2535 \fi
2536 \fi
2537 \sbox\@tempboxa{\makelabel{#1}}%
2538 \global\setbox\@labels\hbox{%
2539 \unhbox\@labels
2540 \hskip \itemindent
2541 \hskip -\labelwidth
2542 \hskip -\labelsep
2543 \ifdim \wd\@tempboxa >\labelwidth
2544 \box\@tempboxa
2545 \else
2546 \hbox to\labelwidth {\unhbox\@tempboxa}%
2547 \fi
2548 \hskip \labelsep}%
2549 \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。

```

2550 \def\@afterheading{%
2551 \@nobreaktrue
2552 \everypar{%
2553 \if@nobreak
2554 \@nobreakfalse
2555 \clubpenalty \@M
2556 \if@afterindent \else
2557 {\setbox\z@\lastbox}%
2558 \fi
2559 \else
2560 \clubpenalty \@clubpenalty
2561 \everypar{\everyparhook}%
2562 \fi\everyparhook}}

```

`\@gnewline` についてはちょっと複雑な心境です。もともとの `pLATEX 2ε` 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。

```

2563 \def\@gnewline #1{%
2564 \ifvmode
2565 \@nolnerr
2566 \else
2567 \unskip \reserved@e {\reserved@f#1}\nobreak \hfil \break \null
2568 \inhibitglue \ignorespaces

```

2569 \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。

```
2570 \if@jslogo
2571   \IfFileExists{jslogo.sty}{%
2572     \RequirePackage{jslogo}%
2573     \def\小{\jslg@small}%
2574     \def\上小{\jslg@uppersmall}%
2575   }{%
2576     \ClassWarningNoLine{\jsc@clsname}{%
2577       The redefinitions of LaTeX-related logos has\MessageBreak
2578       been moved to jslogo.sty since 2016, but\MessageBreak
2579       jslogo.sty not found. Current release of\MessageBreak
2580       'jsclasses' includes it, so please check\MessageBreak
2581       the installation}%
2582   }
2583 \fi
```

13 amsmath との衝突の回避

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

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

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

```
2584 %\let\ltx@ifnextchar\@ifnextchar
2585 %\def\ProvidesFile#1{%
2586 %  \begingroup
2587 %    \catcode'\ 10 %
2588 %    \ifnum \endlinechar<256 %
2589 %      \ifnum \endlinechar>\m@ne
2590 %        \catcode\endlinechar 10 %
2591 %    \fi
```

```

2592 % \fi
2593 % \@makeother\/%
2594 % \@makeother\&%
2595 % \ltx@ifnextchar[{\@providesfile{#1}}{\@providesfile{#1}[]}]

```

14 初期設定

■いろいろな語

```

\prepartname
\postpartname 2596 \newcommand{\prepartname}{\if@english Part~\else 第 \fi}
\prechaptername 2597 \newcommand{\postpartname}{\if@english \else 部 \fi}
2598 (book | report)\newcommand{\prechaptername}{\if@english Chapter~\else 第 \fi}
\postchaptername 2599 (book | report)\newcommand{\postchaptername}{\if@english \else 章 \fi}
\presectionname 2600 \newcommand{\presectionname}{}% 第
\postsectionname 2601 \newcommand{\postsectionname}{}% 節

\contentsname
\listfigurename 2602 \newcommand{\contentsname}{\if@english Contents\else 目次 \fi}
\listtablename 2603 \newcommand{\listfigurename}{\if@english List of Figures\else 図目次 \fi}
2604 \newcommand{\listtablename}{\if@english List of Tables\else 表目次 \fi}

\refname
\bibname 2605 \newcommand{\refname}{\if@english References\else 参考文献 \fi}
\indexname 2606 \newcommand{\bibname}{\if@english Bibliography\else 参考文献 \fi}
2607 \newcommand{\indexname}{\if@english Index\else 索引 \fi}

\figurename
\tablename 2608 (!jspf)\newcommand{\figurename}{\if@english Fig.~\else 図 \fi}
2609 (jspf)\newcommand{\figurename}{Fig.~}
2610 (!jspf)\newcommand{\tablename}{\if@english Table~\else 表 \fi}
2611 (jspf)\newcommand{\tablename}{Table~}

\appendixname
\abstractname 2612 % \newcommand{\appendixname}{\if@english Appendix~\else 付録 \fi}
2613 \newcommand{\appendixname}{\if@english \else 付録 \fi}
2614 (!book)\newcommand{\abstractname}{\if@english Abstract\else 概要 \fi}

\today
2615 \newif\if 西暦 \西暦 true
2616 \def\西暦{\西暦 true}
2617 \def\和暦{\西暦 false}
2618 \newcount\heisei \heisei\year \advance\heisei-1988\relax

```

■今日の日付 L^AT_EX で処理した日付を出力します。jarticle などと違って、標準を西暦にし、余分な空白が入らないように改良しました。和暦にするには `\和暦` と書いてください。

```

2619 \def\today{%
2620   \if@english
2621     \ifcase\month\or
2622       January\or February\or March\or April\or May\or June\or
2623       July\or August\or September\or October\or November\or December\fi
2624     \space\number\day, \number\year
2625   \else
2626     \if 西暦
2627       \number\year 年
2628       \number\month 月
2629       \number\day 日
2630     \else
2631       平成 \number\heisei 年
2632       \number\month 月
2633       \number\day 日
2634     \fi
2635   \fi}

```

■ハイフネーション例外 \TeX のハイフネーションルールの補足です（ペンディング：`eng-lish`）

```
2636 \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。

```

2637 <article>\if@slide \pagestyle{empty} \else \pagestyle{plain} \fi
2638 <book>\if@report \pagestyle{plain} \else \pagestyle{headings} \fi
2639 <report | kiyou>\pagestyle{plain}
2640 <jspf>\pagestyle{headings}
2641 \pagenumbering{arabic}
2642 \if@twocolumn
2643   \twocolumn
2644   \sloppy
2645   \flushbottom
2646 \else
2647   \onecolumn
2648   \raggedbottom
2649 \fi
2650 \if@slide
2651   \renewcommand\kanjifamilydefault{\gtdefault}
2652   \renewcommand\familydefault{\sfdefault}
2653   \raggedright
2654   \xkanjiskip=0.1em\relax
2655 \fi

```

15 実験的コード

[2016-11-29] コミュニティ版 $\text{p}\text{L}\text{A}\text{T}\text{E}\text{X}$ で新設されたテスト用パッケージ（`expp12e` 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。expp12e 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。

```
2656 (*article | book | report)
2657 \@ifpackageloaded{expp12e}{\jsc@needsp@tchtrue}{\jsc@needsp@tchfalse}
2658 \ifjsc@needsp@tch\else
2659   \expandafter\endinput
2660 \fi
```

以下は実験的コードです。具体的には, 2016/11/29 の expp12e 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
2661 \def\@gnewline #1{%
2662   \ifvmode
2663     \@nolnerr
2664   \else
2665     \unskip \reserved@e {\reserved@f#1}\nobreak \hfil \break \hskip \z@
2666     \ignorespaces
2667   \fi}
2668 </article | book | report>
```

以上です。