

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

2020/02/02

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、 $\text{T}_{\text{E}}\text{X}$ の `\mag` プリミティブで全体的に拡大するという手を使って、9 ポイントや 21, 25, 30, 36, 43 ポイント, 12Q, 14Q の指定を可能にしています。

以下では実際のコードに即して説明します。

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

```
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\jpsc@magtrue
69 \newif\ifjpsc@mag@xreal\jpsc@mag@xrealfalse
70 \def\jpsc@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$\displaystyle\tabskip\z@skip{##}$\@eqnsele
137 &\global\@eqcnt\@ne \hfil$\displaystyle{##}\hfil
138 &\global\@eqcnt\tw@ $\displaystyle{##}$\hfil\tabskip\@centering
139 &\global\@eqcnt\thr@@ \hbext\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$\displaystyle\tabskip\z@skip{##}$\@eqnsele
163 &\global\@eqcnt\@ne \hfil$\displaystyle{##}\hfil
164 &\global\@eqcnt\tw@
165 $\displaystyle{##}$\hfil \tabskip\@centering
166 &\global\@eqcnt\thr@@ \hbext\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, dvi2pdf などでも出力出来るようになる) が使えます。

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

```
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 `truein` 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, T_EX 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, pT_EX (アスキーが日本語化した T_EX) の公称 10 ポイントの和文フォント (min10 など) は, 実寸 (標準の字送り量) が 9.62216pt です。これは 3.3818mm, 写研の写植機の単位では 13.527 級, PostScript の単位では 9.5862 ポイントになります。jis フォントなどもこの値を踏襲しています。

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

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

```

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 $\backslash\text{Cjascale}$ 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}{ }

```

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

```

532 \ifx\@rmfamilyhook\@undefined % old
533 \DeclareRobustCommand\rmfamily
534     {\not@math@alphabet\rmfamily\mathrm
535     \romanfamily\rmdefault\kanjifamily\mcdefault\selectfont}
536 \DeclareRobustCommand\sffamily
537     {\not@math@alphabet\sffamily\mathsf
538     \romanfamily\sfdefault\kanjifamily\gtdefault\selectfont}
539 \DeclareRobustCommand\ttfamily
540     {\not@math@alphabet\ttfamily\mathtt
541     \romanfamily\ttdefault\kanjifamily\gtdefault\selectfont}
542 \AtBeginDocument{%
543     \ifx\mweights@init\@undefined\else % mweights.sty is loaded
544     % my definitions above should have been overwritten, recover it!
545     % \selectfont is executed twice but I don't care about speed...
546     \expandafter\g@addto@macro\csname rmfamily \endcsname
547         {\kanjifamily\mcdefault\selectfont}%
548     \expandafter\g@addto@macro\csname sffamily \endcsname

```

```

549     {\kanjifamily\gtdefault\selectfont}%
550     \expandafter\g@addto@macro\csname ttfamily \endcsname
551     {\kanjifamily\gtdefault\selectfont}%
552     \fi}
553 \else                                     % 2020-02-02
554 \g@addto@macro\rmfamilyhook
555     {\prepare@family@series@update@kanji{mc}\mcdefault}
556 \g@addto@macro\sffamilyhook
557     {\prepare@family@series@update@kanji{gt}\gtdefault}
558 \g@addto@macro\ttfamilyhook
559     {\prepare@family@series@update@kanji{gt}\gtdefault}
560 \fi

```

`\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。

```

561 \ifx\DeclareFixJFMCJKTextFontCommand\undefined
562 \DeclareRobustCommand\textmc[1]{%
563     \relax\ifmmode \expandafter\nfss@text \fi{\mcfamily #1}}
564 \DeclareRobustCommand\textgt[1]{%
565     \relax\ifmmode \expandafter\nfss@text \fi{\gtfamily #1}}
566 \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。

```

567 \def\reDeclareMathAlphabet#1#2#3{%
568     \edef\@tempa{\expandafter@gobble\string#2}%
569     \edef\@tempb{\expandafter@gobble\string#3}%
570     \edef\@tempc{\string @\expandafter@gobbletwo\string#2}%
571     \ifx\@tempc\@tempa%

```

```

572 \edef\@tempaf\expandafter\@gobbletwo\string#2}%
573 \edef\@tempbf\expandafter\@gobbletwo\string#3}%
574 \fi
575 \beginingroup
576 \let\protect\noexpand
577 \def\@tempaaf\relax}%
578 \expandafter\ifx\csname RDMAorg@\@tempa\endcsname\relax
579 \edef\@tempaa{\expandafter\def\expandafter\noexpand%
580 \csname RDMAorg@\@tempa\endcsname{%
581 \expandafter\noexpand\csname\@tempa\endcsname}}%
582 \fi
583 \def\@tempbb{\relax}%
584 \expandafter\ifx\csname RDMAorg@\@tempb\endcsname\relax
585 \edef\@tempbb{\expandafter\def\expandafter\noexpand%
586 \csname RDMAorg@\@tempb\endcsname{%
587 \expandafter\noexpand\csname\@tempb\endcsname}}%
588 \fi
589 \edef\@tempc{\@tempaa\@tempbb}%
590 \expandafter\endgroup\@tempc%
591 \edef#1{\noexpand\protect\expandafter\noexpand\csname%
592 \expandafter\@gobble\string#1\space\space\endcsname}%
593 \expandafter\edef\csname\expandafter\@gobble\string#1\space\space\endcsname%
594 {\noexpand\DualLang@mathalph@bet%
595 {\expandafter\noexpand\csname RDMAorg@\@tempa\endcsname}}%
596 {\expandafter\noexpand\csname RDMAorg@\@tempb\endcsname}}%
597 }%
598 }
599 \@onlypreamble\reDeclareMathAlphabet
600 \def\DualLang@mathalph@bet#1#2{%
601 \relax\ifmmode
602 \ifx\math@bgroup\bgroup% 2e normal style (\mathrm{...})
603 \bgroup\let\DualLang@Mfontsw\DLMfontsw@standard
604 \else
605 \ifx\math@bgroup\relax% 2e two letter style (\rm->\mathrm)
606 \let\DualLang@Mfontsw\DLMfontsw@oldstyle
607 \else
608 \ifx\math@bgroup\@empty% 2.09 oldfont style ({\mathrm ...})
609 \let\DualLang@Mfontsw\DLMfontsw@oldfont
610 \else% panic! assume 2e normal style
611 \bgroup\let\DualLang@Mfontsw\DLMfontsw@standard
612 \fi
613 \fi
614 \fi
615 \else
616 \let\DualLang@Mfontsw\@firstoftwo
617 \fi
618 \DualLang@Mfontsw{#1}{#2}%
619 }
620 \def\DLMfontsw@standard#1#2#3{#1{#2{#3}}\egroup}

```

```

621 \def\DLMfontsw@oldstyle#1#2{#1\relax\@fontswitch\relax{#2}}
622 \def\DLMfontsw@oldfont#1#2{#1\relax#2\relax}
623 \if@enablejfam
624   \DeclareSymbolFont{mincho}{\jsc@JYn}{mc}{m}{n}
625   \DeclareSymbolFontAlphabet{\mathmc}{mincho}
626   \SetSymbolFont{mincho}{bold}{\jsc@JYn}{gt}{m}{n}
627   \jfam\symmincho
628   \DeclareMathAlphabet{\mathgt}{\jsc@JYn}{gt}{m}{n}
629   \AtBeginDocument{%
630     \reDeclareMathAlphabet{\mathrm}{\@mathrm}{\@mathmc}
631     \reDeclareMathAlphabet{\mathbf}{\@mathbf}{\@mathgt}}
632 \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。

```
633 % \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。

```

634 \prebreakpenalty\jis"2147=10000      % 5000   '
635 \postbreakpenalty\jis"2148=10000     % 5000   "
636 \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。

```

637 \inhibitxspcode'!=1
638 \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

```
639 % \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。

```

640 \xspcode'+=3
641 \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。

642 \xspcode'^^80=3
643 \xspcode'^^81=3
644 \xspcode'^^82=3
645 \xspcode'^^83=3
646 \xspcode'^^84=3
647 \xspcode'^^85=3
648 \xspcode'^^86=3
649 \xspcode'^^87=3
650 \xspcode'^^88=3
651 \xspcode'^^89=3
652 \xspcode'^^8a=3
653 \xspcode'^^8b=3
654 \xspcode'^^8c=3
655 \xspcode'^^8d=3
656 \xspcode'^^8e=3
657 \xspcode'^^8f=3
658 \xspcode'^^90=3
659 \xspcode'^^91=3
660 \xspcode'^^92=3
661 \xspcode'^^93=3
662 \xspcode'^^94=3
663 \xspcode'^^95=3
664 \xspcode'^^96=3
665 \xspcode'^^97=3
666 \xspcode'^^98=3
667 \xspcode'^^99=3
668 \xspcode'^^9a=3
669 \xspcode'^^9b=3
670 \xspcode'^^9c=3
671 \xspcode'^^9d=3
672 \xspcode'^^9e=3
673 \xspcode'^^9f=3
674 \xspcode'^^a0=3
675 \xspcode'^^a1=3
676 \xspcode'^^a2=3
677 \xspcode'^^a3=3
678 \xspcode'^^a4=3
679 \xspcode'^^a5=3
680 \xspcode'^^a6=3
681 \xspcode'^^a7=3
682 \xspcode'^^a8=3
683 \xspcode'^^a9=3
684 \xspcode'^^aa=3
685 \xspcode'^^ab=3
686 \xspcode'^^ac=3
687 \xspcode'^^ad=3
688 \xspcode'^^ae=3
689 \xspcode'^^af=3
690 \xspcode'^^b0=3

691 \xspcode'^b1=3
692 \xspcode'^b2=3
693 \xspcode'^b3=3
694 \xspcode'^b4=3
695 \xspcode'^b5=3
696 \xspcode'^b6=3
697 \xspcode'^b7=3
698 \xspcode'^b8=3
699 \xspcode'^b9=3
700 \xspcode'^ba=3
701 \xspcode'^bb=3
702 \xspcode'^bc=3
703 \xspcode'^bd=3
704 \xspcode'^be=3
705 \xspcode'^bf=3
706 \xspcode'^c0=3
707 \xspcode'^c1=3
708 \xspcode'^c2=3
709 \xspcode'^c3=3
710 \xspcode'^c4=3
711 \xspcode'^c5=3
712 \xspcode'^c6=3
713 \xspcode'^c7=3
714 \xspcode'^c8=3
715 \xspcode'^c9=3
716 \xspcode'^ca=3
717 \xspcode'^cb=3
718 \xspcode'^cc=3
719 \xspcode'^cd=3
720 \xspcode'^ce=3
721 \xspcode'^cf=3
722 \xspcode'^d0=3
723 \xspcode'^d1=3
724 \xspcode'^d2=3
725 \xspcode'^d3=3
726 \xspcode'^d4=3
727 \xspcode'^d5=3
728 \xspcode'^d6=3
729 \xspcode'^d7=3
730 \xspcode'^d8=3
731 \xspcode'^d9=3
732 \xspcode'^da=3
733 \xspcode'^db=3
734 \xspcode'^dc=3
735 \xspcode'^dd=3
736 \xspcode'^de=3
737 \xspcode'^df=3
738 \xspcode'^e0=3
739 \xspcode'^e1=3

```

740 \xspcode{^e2=3
741 \xspcode{^e3=3
742 \xspcode{^e4=3
743 \xspcode{^e5=3
744 \xspcode{^e6=3
745 \xspcode{^e7=3
746 \xspcode{^e8=3
747 \xspcode{^e9=3
748 \xspcode{^ea=3
749 \xspcode{^eb=3
750 \xspcode{^ec=3
751 \xspcode{^ed=3
752 \xspcode{^ee=3
753 \xspcode{^ef=3
754 \xspcode{^f0=3
755 \xspcode{^f1=3
756 \xspcode{^f2=3
757 \xspcode{^f3=3
758 \xspcode{^f4=3
759 \xspcode{^f5=3
760 \xspcode{^f6=3
761 \xspcode{^f7=3
762 \xspcode{^f8=3
763 \xspcode{^f9=3
764 \xspcode{^fa=3
765 \xspcode{^fb=3
766 \xspcode{^fc=3
767 \xspcode{^fd=3
768 \xspcode{^fe=3
769 \xspcode{^ff=3

```

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

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

```
770 \def\@{\spacefactor3000{}}
```

4 フォントサイズ

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

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

のようにして行います。これは

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

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

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

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

`\kanjiskip` は pL^AT_EX 2_ε で `0pt plus .4pt minus .5pt` に設定していますが、これはそもそも文字サイズの変更に応じて変わるべきものです。それに、プラスになったりマイナスになったりするのには、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。

```
771 \def\@setfontsize#1#2#3{%
772 % \@nomath#1%
773 \ifx\protect\@typeset@protect
774 \let\@currszsize#1%
775 \fi
776 \fontsize{#2}{#3}\selectfont
777 \ifdim\parindent>\z@
778 \if@english
779 \parindent=1em
780 \else
781 \parindent=1zw
782 \fi
783 \fi
784 \kanjiskip=0zw plus .1zw minus .01zw
785 \ifdim\xkanjiskip>\z@
786 \if@slide \xkanjiskip=0.1em \else
787 \xkanjiskip=0.25em plus 0.15em minus 0.06em
788 \fi
789 \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。

```
790 \def\jsc@setfontsize#1#2#3{%
```

```
791 \@setfontsize#1{#2\jsc@mp}{#3\jsc@mp}}
```

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。

```
792 \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

`\widebaselines` `\abovedisplayskip` 等が初期化されてしまうという shintok さんのご指摘に対し

て、しばしば愛好家さんが次の修正を教えてくださいました。

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

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

```
793 \newif\ifnarrowbaselines
```

```
794 \if@english
```

```
795 \narrowbaselinestrue
```

```
796 \fi
```

```
797 \def\narrowbaselines{%
```

```
798 \narrowbaselinestrue
```

```
799 \skip0=\abovedisplayskip
```

```
800 \skip2=\abovedisplayshortskip
```

```
801 \skip4=\belowdisplayskip
```

```
802 \skip6=\belowdisplayshortskip
```

```
803 \@currsize\selectfont
```

```
804 \abovedisplayskip=\skip0
```

```
805 \abovedisplayshortskip=\skip2
```

```
806 \belowdisplayskip=\skip4
```

```
807 \belowdisplayshortskip=\skip6\relax}
```

```
808 \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。

```
809 \renewcommand{\normalsize}{%
```

```
810 \ifnarrowbaselines
```

```
811 \jsc@setfontsize\normalsize\@xpt\@xipt
```

```
812 \else
```

```
813 \jsc@setfontsize\normalsize\@xpt{\n@baseline}%
```

```
814 \fi
```

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

[2003-02-16] ちょっと変えました。

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

```
815 \abovedisplayskip 11\jsc@empt \@plus3\jsc@empt \@minus4\jsc@empt
816 \abovedisplayshortskip \z@ \@plus3\jsc@empt
817 \belowdisplayskip 9\jsc@empt \@plus3\jsc@empt \@minus4\jsc@empt
818 \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。

```
819 \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。

```
820 \normalsize
```

`\Cht` 基準となる長さの設定をします。pL^AT_EX 2_ε カーネル (`plfonts.dtx`) で宣言されているパラメータに実際の値を設定します。たとえば `\Cwd` は `\normalfont` の全角幅 (`1zw`) です。
`\Cwd` [2017-08-31] 基準とする文字を「全角空白」(EUC コード `0xA1A1`) から「漢」(JIS コード `0x3441`) へ変更しました。
`\Cvs` [2017-09-19] 内部的に使った `\box0` を空にします。
`\Chs` [2017-09-19] 内部的に使った `\box0` を空にします。

```
821 \setbox0\hbox{\char\jis"3441}%
822 \setlength\Cht{\ht0}
823 \setlength\Cdp{\dp0}
824 \setlength\Cwd{\wd0}
825 \setlength\Cvs{\baselineskip}
826 \setlength\Chs{\wd0}
827 \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。

```
828 \newcommand{\small}{%
829   \ifnarrowbaselines
830     (!kiyou)   \jsc@setfontsize\small\@ixpt{11}%
831     (kiyou)   \jsc@setfontsize\small{8.8888}{11}%
832   \else
833     (!kiyou)   \jsc@setfontsize\small\@ixpt{13}%
834     (kiyou)   \jsc@setfontsize\small{8.8888}{13.2418}%
835   \fi
836 \abovedisplayskip 9\jsc@empt \@plus3\jsc@empt \@minus4\jsc@empt
837 \abovedisplayshortskip \z@ \@plus3\jsc@empt
838 \belowdisplayskip \abovedisplayskip
839 \belowdisplayshortskip \belowdisplayskip
840 \def\@listi{\leftmargin\leftmargini
841             \topsep \z@
842             \parsep \z@
843             \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。

```

844 \newcommand{\footnotesize}{%
845   \ifnarrowbaselines
846   (!kiyou)   \jsc@setfontsize\footnotesize\@viiipt{9.5}%
847   (kiyou)    \jsc@setfontsize\footnotesize{8.8888}{11}%
848   \else
849   (!kiyou)   \jsc@setfontsize\footnotesize\@viiipt{11}%
850   (kiyou)    \jsc@setfontsize\footnotesize{8.8888}{13.2418}%
851   \fi
852   \abovedisplayskip 6\jsc@mpt \@plus2\jsc@mpt \@minus3\jsc@mpt
853   \abovedisplayshortskip \z@ \@plus2\jsc@mpt
854   \belowdisplayskip \abovedisplayskip
855   \belowdisplayshortskip \belowdisplayskip
856   \def\@listi{\leftmargin\leftmargini
857             \topsep \z@
858             \parsep \z@
859             \itemsep \parsep}}

```

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

`\LARGE` [2004-11-03] `\HUGE` を追加。

```

\huge 860 \newcommand{\scriptsize}{\jsc@setfontsize\scriptsize\@viipt\@viipt}
861 \newcommand{\tiny}{\jsc@setfontsize\tiny\@vpt\@vpt}
\Lhuge 862 \if@twocolumn
\LHUGE 863 (!kiyou) \newcommand{\large}{\jsc@setfontsize\large\@xiipt{\n@baseline}}
864 (kiyou) \newcommand{\large}{\jsc@setfontsize\large{11.111}{\n@baseline}}
865 \else
866 (!kiyou) \newcommand{\large}{\jsc@setfontsize\large\@xiipt{17}}
867 (kiyou) \newcommand{\large}{\jsc@setfontsize\large{11.111}{17}}
868 \fi
869 (!kiyou)\newcommand{\LARGE}{\jsc@setfontsize\LARGE\@xivpt{21}}
870 (kiyou)\newcommand{\LARGE}{\jsc@setfontsize\LARGE{12.222}{21}}
871 \newcommand{\LARGE}{\jsc@setfontsize\LARGE\@xviipt{25}}
872 \newcommand{\huge}{\jsc@setfontsize\huge\@xxpt{28}}
873 \newcommand{\Huge}{\jsc@setfontsize\Huge\@xxvpt{33}}
874 \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。

```

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

```

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

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

```
876 % \newcommand{\headfont}{\bfseries}
877 \newcommand{\headfont}{\gtfamily\sffamily}
878 % \newcommand{\headfont}{\sffamily\fontseries{sbc}\selectfont}
```

5 レイアウト

■二段組

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

```
879 (!kiyou)\setlength\columnsep{2zw}
880 (kiyou)\setlength\columnsep{28truebp}
881 \setlength\columnseprule{\z@}
```

■段落

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

```
\lineskiplimit 882 \setlength\lineskip{1\jsc@empt}
\normallineskip 883 \setlength\normallineskip{1\jsc@empt}
\normallineskiplimit 884 \setlength\lineskiplimit{1\jsc@empt}
885 \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。

```
886 \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。

```
887 \setlength\parskip{\z@}
888 \if@slide
889 \setlength\parindent{0zw}
```

```

890 \else
891 \setlength\parindent{1zw}
892 \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 893 \@lowpenalty 51
894 \@medpenalty 151
895 \@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。

```
896 % \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。

```
897 % \brokenpenalty 100
```

5.1 ページレイアウト

■縦方向のスペース

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

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

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

```

898 \setlength\topskip{1.38zw}%% from 10\jsc@mpt (2016-08-17)
899 \if@slide
900 \setlength\headheight{0\jsc@mpt}
901 \else
902 \setlength\headheight{20\jsc@mpt}%% from 2\topskip (2016-08-17); from \topskip (2003-06-26)
903 \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。

```

904 (*article|kiyou)
905 \if@slide
906 \setlength\footskip{0pt}
907 \else
908 \setlength\footskip{0.03367\paperheight}

```

```

909 \ifdim\footskip<\baselineskip
910   \setlength\footskip{\baselineskip}
911 \fi
912 \fi
913 </article | kiyou>
914 <jspf>\setlength\footskip{9\jsc@mmm}
915 <*book>
916 \if@report
917   \setlength\footskip{0.03367\paperheight}
918   \ifdim\footskip<\baselineskip
919     \setlength\footskip{\baselineskip}
920   \fi
921 \else
922   \setlength\footskip{0pt}
923 \fi
924 </book>
925 <*report>
926 \setlength\footskip{0.03367\paperheight}
927 \ifdim\footskip<\baselineskip
928   \setlength\footskip{\baselineskip}
929 \fi
930 </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)。

```

931 <*article>
932 \if@slide
933   \setlength\headsep{0\jsc@mpt}
934   \addtolength\headsep{-\topskip}%% added (2016-10-08)
935   \addtolength\headsep{10\jsc@mpt}%% added (2016-10-08)
936 \else
937   \setlength\headsep{\footskip}
938   \addtolength\headsep{-\topskip}
939 \fi
940 </article>
941 <*book>
942 \if@report
943   \setlength\headsep{\footskip}
944   \addtolength\headsep{-\topskip}
945 \else
946   \setlength\headsep{6\jsc@mmm}
947   \addtolength\headsep{-\topskip}%% added (2016-10-08)
948   \addtolength\headsep{10\jsc@mpt}%% added (2016-10-08)
949 \fi
950 </book>

```

```

951 <*report>
952 \setlength\headsep{\footskip}
953 \addtolength\headsep{-\topskip}
954 </report>
955 <*jspf>
956 \setlength\headsep{9\jsc@mmm}
957 \addtolength\headsep{-\topskip}
958 </jspf>
959 <*kiyou>
960 \setlength\headheight{0\jsc@mpt}
961 \setlength\headsep{0\jsc@mpt}
962 \addtolength\headsep{-\topskip}%% added (2016-10-08)
963 \addtolength\headsep{10\jsc@mpt}%% added (2016-10-08)
964 </kiyou>

```

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

```
965 \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。

```
966 \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。

```

967 <*article>
968 \if@slide
969   \setlength\fullwidth{0.9\paperwidth}
970 \else
971   \setlength\fullwidth{0.76\paperwidth}
972 \fi
973 \if@twocolumn \@tempdima=2zw \else \@tempdima=1zw \fi
974 \divide\fullwidth\@tempdima \multiply\fullwidth\@tempdima
975 \setlength\textwidth{\fullwidth}
976 </article>

```



```

977 (*book)
978 \if@report
979 \setlength\fullwidth{0.76\paperwidth}
980 \else
981 \setlength\fullwidth{\paperwidth}
982 \addtolength\fullwidth{-36\jsc@mmm}
983 \fi
984 \if@twocolumn \@tempdima=2zw \else \@tempdima=1zw \fi
985 \divide\fullwidth\@tempdima \multiply\fullwidth\@tempdima
986 \setlength\textwidth{\fullwidth}
987 \if@report \else
988 \if@twocolumn \else
989 \ifdim \fullwidth>40zw
990 \setlength\textwidth{40zw}
991 \fi
992 \fi
993 \fi
994 (/book)
995 (*report)
996 \setlength\fullwidth{0.76\paperwidth}
997 \if@twocolumn \@tempdima=2zw \else \@tempdima=1zw \fi
998 \divide\fullwidth\@tempdima \multiply\fullwidth\@tempdima
999 \setlength\textwidth{\fullwidth}
1000 (/report)
1001 (*jspf)
1002 \setlength\fullwidth{50zw}
1003 \addtolength\fullwidth{8\jsc@mmm}
1004 \setlength\textwidth{\fullwidth}
1005 (/jspf)
1006 (*kiyou)
1007 \setlength\fullwidth{48zw}
1008 \addtolength\fullwidth{\columnsep}
1009 \setlength\textwidth{\fullwidth}
1010 (/kiyou)

```

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

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

某学会誌スタイルでは 44 行にします。

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

[2016-08-26] `\topskip` を 10pt から 1.38zw に増やしましたので、その分 `\textheight`

を増やします (2016-08-17 での修正漏れ)。

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

```
1011 (*article | book | report)
1012 \if@slide
1013 \setlength{\textheight}{0.95\paperheight}
1014 \else
1015 \setlength{\textheight}{0.83\paperheight}
1016 \fi
1017 \addtolength{\textheight}{-10\jsc@mpt}%% from -\topskip (2016-10-08); from -\headheight (2003-
1018 \addtolength{\textheight}{-\headsep}
1019 \addtolength{\textheight}{-\footskip}
1020 \addtolength{\textheight}{-\topskip}
1021 \divide\textheight\baselineskip
1022 \multiply\textheight\baselineskip
1023 </article | book | report)
1024 (jspf)\setlength{\textheight}{51\baselineskip}
1025 (kiyou)\setlength{\textheight}{47\baselineskip}
1026 \addtolength{\textheight}{\topskip}
1027 \addtolength{\textheight}{0.1\jsc@mpt}
1028 (jspf)\setlength{\mathindent}{10\jsc@mmm}
```

`\flushbottom` [2016-07-18] `\textheight` に念のため 0.1 ポイント余裕を持たせているのと同様に, `\flushbottom` にも余裕を持たせます。元の L^AT_EX 2_ε 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。

```
1029 \def\flushbottom{%
1030 \def\@textbottom{\vskip \z@ \@plus.1\jsc@mpt}%
1031 \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。

```
1032 \setlength\marginparsep{\columnsep}
1033 \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。T_EX 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 pL^AT_EX 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。

```
1034 \setlength{\oddsidemargin}{\paperwidth}
1035 \addtolength{\oddsidemargin}{-\fullwidth}
1036 \setlength{\oddsidemargin}{.5\oddsidemargin}
1037 \iftombow
1038 \addtolength{\oddsidemargin}{-1in}
```

```

1039 \else
1040 \addtolength{\oddsidemargin}{-\inv@mag in}
1041 \fi
1042 \setlength{\evensidemargin}{\oddsidemargin}
1043 \if@mparswitch
1044 \addtolength{\evensidemargin}{\fullwidth}
1045 \addtolength{\evensidemargin}{-\textwidth}
1046 \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。

```

1047 \setlength\marginparwidth{\paperwidth}
1048 \addtolength\marginparwidth{-\oddsidemargin}
1049 \addtolength\marginparwidth{-\inv@mag in}
1050 \addtolength\marginparwidth{-\textwidth}
1051 \addtolength\marginparwidth{-10\jsc@mmm}
1052 \addtolength\marginparwidth{-\marginparsep}
1053 \@tempdima=1zw
1054 \divide\marginparwidth\@tempdima
1055 \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)。

```

1056 \setlength\topmargin{\paperheight}
1057 \addtolength\topmargin{-\textheight}
1058 \if@slide
1059 \addtolength\topmargin{-\headheight}
1060 \else
1061 \addtolength\topmargin{-10\jsc@mpt}%% from -\topskip (2016-10-08); from -\headheight (2003-0
1062 \fi
1063 \addtolength\topmargin{-\headsep}
1064 \addtolength\topmargin{-\footskip}
1065 \setlength\topmargin{0.5\topmargin}
1066 (kiyou)\setlength\topmargin{81truebp}
1067 \iftombow
1068 \addtolength\topmargin{-1in}
1069 \else
1070 \addtolength\topmargin{-\inv@mag in}
1071 \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。

```
1072 {\footnotesize\global\setlength\footnotesep{\baselineskip}}
1073 \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。

```
1074 \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。

```
1075 \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。

```
1076 \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。

```
1077 \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。

```
1078 \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。

```
1079 \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。

```
1080 \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。

```
1081 \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。

```
1082 \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。

```
1083 \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。

```
1084 \renewcommand{\dblfloatpagefraction}{.8}
```

`\floatsep` `\floatsep` はページ上部・下部のフロート間の距離です。`\textfloatsep` はページ上部・
`\textfloatsep` 下部のフロートと本文との距離です。`\intextsep` は本文の途中に出力されるフロートと本
`\intextsep` 文との距離です。

```
1085 \setlength\floatsep {12\jsc@empt \@plus 2\jsc@empt \@minus 2\jsc@empt}
```

```
1086 \setlength\textfloatsep{20\jsc@empt \@plus 2\jsc@empt \@minus 4\jsc@empt}
```

```
1087 \setlength\intextsep {12\jsc@empt \@plus 2\jsc@empt \@minus 2\jsc@empt}
```

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

```
\dbltextfloatsep 1088 \setlength\dblfloatsep {12\jsc@empt \@plus 2\jsc@empt \@minus 2\jsc@empt}
```

```
1089 \setlength\dbltextfloatsep{20\jsc@empt \@plus 2\jsc@empt \@minus 4\jsc@empt}
```

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

`\@fpsep` `\@fpsep` はフロート間に入ります。

```
\@fpbot 1090 \setlength\@fptop{0\jsc@empt \@plus 1fil}
```

```
1091 \setlength\@fpsep{8\jsc@empt \@plus 2fil}
```

```
1092 \setlength\@fpbot{0\jsc@empt \@plus 1fil}
```

`\@dblfpsep` 段抜きフロートについての値です。

```
\@dblfpsep 1093 \setlength\@dblfpsep{0\jsc@empt \@plus 1fil}
```

```
\@dblfpbot 1094 \setlength\@dblfpsep{8\jsc@empt \@plus 2fil}
```

```
1095 \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_LA_TE_X の標準クラス 2017/02/15 に合わせて, 同じ命令を追
`\pltx@cleartoleftpage` 加しました。

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

となっています。

```
1096 (*article | book | report)
```

```
1097 \def\pltx@cleartorightpage{\clearpage\if@twoside
```

```
1098 \ifodd\c@page
```

```
1099 \iftdir
```

```
1100 \hbox{ }\thispagestyle{empty}\newpage
```

```

1101     \if@twocolumn\hbox{}\newpage\fi
1102     \fi
1103     \else
1104     \ifydir
1105     \hbox{}\thispagestyle{empty}\newpage
1106     \if@twocolumn\hbox{}\newpage\fi
1107     \fi
1108     \fi\fi}
1109 \def\pltx@cleartoleftpage{\clearpage\if@twoside
1110 \ifodd\c@page
1111     \ifydir
1112     \hbox{}\thispagestyle{empty}\newpage
1113     \if@twocolumn\hbox{}\newpage\fi
1114     \fi
1115     \else
1116     \iftdir
1117     \hbox{}\thispagestyle{empty}\newpage
1118     \if@twocolumn\hbox{}\newpage\fi
1119     \fi
1120     \fi\fi}
1121 \def\pltx@cleartooddpage{\clearpage\if@twoside
1122 \ifodd\c@page\else
1123     \hbox{}\thispagestyle{empty}\newpage
1124     \if@twocolumn\hbox{}\newpage\fi
1125     \fi\fi}
1126 \def\pltx@cleartoevenpage{\clearpage\if@twoside
1127 \ifodd\c@page
1128     \hbox{}\thispagestyle{empty}\newpage
1129     \if@twocolumn\hbox{}\newpage\fi
1130     \fi\fi}
1131 </article | book | report>

```

`\cleardoublepage` [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、`report` と `book` クラスの場合に `\cleardoublepage` を再定義します。

```

1132 <*book | report>
1133 \if@openleft
1134     \let\cleardoublepage\pltx@cleartoleftpage
1135 \else\if@openright
1136     \let\cleardoublepage\pltx@cleartorightpage
1137 \fi\fi
1138 </book | report>

```

7 ページスタイル

ページスタイルとして、L^AT_EX 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 L^AT_EX 2_ε 本体で定義されています。

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

ページスタイルは `\ps@...` の形のマクロで定義されています。

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

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

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

柱を設定する命令は、右の柱が左の柱の下位にある場合は十分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。L^AT_EX 本体で定義されているものをコメントアウトした形で載せておきます。

```
1139 % \def\ps@empty{%
1140 %   \let\@mkboth\@gobbletwo
1141 %   \let\@oddhead\@empty
1142 %   \let\@oddfoot\@empty
1143 %   \let\@evenhead\@empty
1144 %   \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。

```
1145 \def\ps@plainfoot{%
1146   \let\@mkboth\@gobbletwo
1147   \let\@oddhead\@empty
1148   \def\@oddfoot{\normalfont\hfil\thepage\hfil}%
1149   \let\@evenhead\@empty
1150   \let\@evenfoot\@oddfoot}
1151 \def\ps@plainhead{%
1152   \let\@mkboth\@gobbletwo
1153   \let\@oddfoot\@empty
1154   \let\@evenfoot\@empty
1155   \def\@evenhead{%
1156     \if@mparswitch \hss \fi
```

```

1157 \hbox to \fullwidth{\textbf{\thepage}\hfil}%
1158 \if@mparswitch\else \hss \fi}%
1159 \def\@oddhead{%
1160 \hbox to \fullwidth{\hfil\textbf{\thepage}}\hss}}
1161 (book)\if@report \let\ps@plain\ps@plainfoot \else \let\ps@plain\ps@plainhead \fi
1162 (!book)\let\ps@plain\ps@plainfoot

```

`\ps@headings` `headings` スタイルはヘッダに見出しとページ番号を出力します。ここではヘッダにアンダーラインを引くようにしてみました。

まず `article` の場合です。

```

1163 (*article | kiyou)
1164 \if@twoside
1165 \def\ps@headings{%
1166 \let\@oddfoot\@empty
1167 \let\@evenfoot\@empty
1168 \def\@evenhead{\if@mparswitch \hss \fi
1169 \underline{\hbox to \fullwidth{\textbf{\thepage}\hfil\leftmark}}}%
1170 \if@mparswitch\else \hss \fi}%
1171 \def\@oddhead{%
1172 \underline{%
1173 \hbox to \fullwidth{{\rightmark}\hfil\textbf{\thepage}}}\hss}%
1174 \let\@mkboth\markboth
1175 \def\sectionmark##1{\markboth{%
1176 \ifnum \c@secnumdepth >\z@ \thesection \hskip1zw\fi
1177 ##1}}}%
1178 \def\subsectionmark##1{\markright{%
1179 \ifnum \c@secnumdepth >\@ne \thesubsection \hskip1zw\fi
1180 ##1}}}%
1181 }
1182 \else % if not twoside
1183 \def\ps@headings{%
1184 \let\@oddfoot\@empty
1185 \def\@oddhead{%
1186 \underline{%
1187 \hbox to \fullwidth{{\rightmark}\hfil\textbf{\thepage}}}\hss}%
1188 \let\@mkboth\markboth
1189 \def\sectionmark##1{\markright{%
1190 \ifnum \c@secnumdepth >\z@ \thesection \hskip1zw\fi
1191 ##1}}}%
1192 \fi
1193 (/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)。

```

1194 (*book | report)
1195 \newif\if@omit@number
1196 \def\ps@headings{%
1197 \let\@oddfoot\@empty
1198 \let\@evenfoot\@empty

```



```

1199 \def\@evenhead{%
1200   \if@mparswitch \hss \fi
1201   \underline{\hbox to \fullwidth{\autoxspacing
1202     \textbf{\thepage}\hfil\leftmark}}}%
1203   \if@mparswitch\else \hss \fi}%
1204 \def\@oddhead{\underline{\hbox to \fullwidth{\autoxspacing
1205   {\if@twoside\rightmark\else\leftmark\fi}\hfil\textbf{\thepage}}}\hss}%
1206 \let\@mkboth\markboth
1207 \def\chaptermark##1{\markboth{%
1208   \ifnum \c@secnumdepth >\m@ne
1209 (book)     \if@mainmatter
1210           \if@omit@number\else
1211           \@chapapp\thechapter\@chappos\hskip1zw
1212           \fi
1213 (book)     \fi
1214   \fi
1215   ##1}{}}%
1216 \def\sectionmark##1{\markright{%
1217   \ifnum \c@secnumdepth >\z@ \thesection \hskip1zw\fi
1218   ##1}}}%
1219 </book | report)

```

最後は学会誌の場合です。

```

1220 (*jspf)
1221 \def\ps@headings{%
1222   \def\@oddfoot{\normalfont\hfil\thepage\hfil}
1223   \def\@evenfoot{\normalfont\hfil\thepage\hfil}
1224   \def\@oddhead{\normalfont\hfil \@title \hfil}
1225   \def\@evenhead{\normalfont\hfil プラズマ・核融合学会誌 \hfil}}
1226 </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。

```

1227 \def\ps@myheadings{%
1228   \let\@oddfoot\@empty\let\@evenfoot\@empty
1229   \def\@evenhead{%
1230     \if@mparswitch \hss \fi%
1231     \hbox to \fullwidth{\thepage\hfil\leftmark}}%
1232     \if@mparswitch\else \hss \fi}%
1233   \def\@oddhead{%
1234     \hbox to \fullwidth{\rightmark\hfil\thepage}\hss}%
1235   \let\@mkboth\@gobbletwo
1236 (book | report) \let\chaptermark\@gobble
1237 \let\sectionmark\@gobble
1238 (!book&!report) \let\subsectionmark\@gobble
1239 }

```

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 1240 % \newcommand*{\title}[1]{\gdef\@title{#1}}
      1241 % \newcommand*{\author}[1]{\gdef\@author{#1}}
      1242 % \newcommand*{\date}[1]{\gdef\@date{#1}}
      1243 % \date{\today}
```

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

```
\eauthor 1244 <*jspf>
1245 \newcommand*{\etitle}[1]{\gdef\@etitle{#1}}
\keywords 1246 \newcommand*{\eauthor}[1]{\gdef\@eauthor{#1}}
          1247 \newcommand*{\keywords}[1]{\gdef\@keywords{#1}}
          1248 \newcommand*{\email}[1]{\gdef\authors@mail{#1}}
          1249 \newcommand*{\AuthorsEmail}[1]{\gdef\authors@mail{author's e-mail:\ #1}}
          1250 </jspf>
```

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

```
1251 \def\plainifnotempty{%
1252   \ifx \@oddhead \@empty
1253     \ifx \@oddfont \@empty
1254       \else
1255         \thispagestyle{plainfoot}%
1256       \fi
1257     \else
1258       \thispagestyle{plainhead}%
1259     \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。

```
1260 <*article | book | report | kiyou>
1261 \if@titlepage
1262   \newcommand{\maketitle}{%
1263     \begin{titlepage}%
1264       \let\footnotesize\small
```

```

1265     \let\footnoterule\relax
1266     \let\footnote\thanks
1267     \null\vfil
1268     \if@slide
1269         {\footnotesize \@date}%
1270         \begin{center}
1271             \mbox{} \\\[1zw]
1272             \large
1273             {\maybeblue\hrule height0\jsc@mpt depth2\jsc@mpt\relax}\par
1274             \jsc@smallskip
1275             \@title
1276             \jsc@smallskip
1277             {\maybeblue\hrule height0\jsc@mpt depth2\jsc@mpt\relax}\par
1278             \vfill
1279             {\small \@author}%
1280         \end{center}
1281     \else
1282     \vskip 60\jsc@mpt
1283     \begin{center}%
1284         {\LARGE \@title \par}%
1285         \vskip 3em%
1286         {\large
1287             \lineskip .75em
1288             \begin{tabular}[t]{c}%
1289                 \@author
1290             \end{tabular}\par}%
1291         \vskip 1.5em
1292         {\large \@date \par}%
1293     \end{center}%
1294     \fi
1295     \par
1296     \@thanks\vfil\null
1297 \end{titlepage}%
1298 \setcounter{footnote}{0}%
1299 \global\let\thanks\relax
1300 \global\let\maketitle\relax
1301 \global\let\@thanks\@empty
1302 \global\let\@author\@empty
1303 \global\let\@date\@empty
1304 \global\let\@title\@empty
1305 \global\let\title\relax
1306 \global\let\author\relax
1307 \global\let\date\relax
1308 \global\let\and\relax
1309 }%
1310 \else
1311     \newcommand{\maketitle}{\par
1312         \begingroup
1313             \renewcommand\thefootnote{\@fnsymbol\c@footnote}%

```

```

1314 \def\makefnmark{\rlap{\@textsuperscript{\normalfont\@thefnmark}}}%
1315 \long\def\makefntext##1{\advance\leftskip 3zw
1316 \parindent 1zw\noindent
1317 \llap{\@textsuperscript{\normalfont\@thefnmark}\hskip0.3zw}##1}%
1318 \if@twocolumn
1319 \ifnum \col@number=\@ne
1320 \maketitle
1321 \else
1322 \twocolumn[\maketitle]%
1323 \fi
1324 \else
1325 \newpage
1326 \global\@topnum\z@ % Prevents figures from going at top of page.
1327 \maketitle
1328 \fi
1329 \plainifnotempty
1330 \@thanks
1331 \endgroup
1332 \setcounter{footnote}{0}%
1333 \global\let\thanks\relax
1334 \global\let\maketitle\relax
1335 \global\let\@thanks\@empty
1336 \global\let\@author\@empty
1337 \global\let\@date\@empty
1338 \global\let\@title\@empty
1339 \global\let\title\relax
1340 \global\let\author\relax
1341 \global\let\date\relax
1342 \global\let\and\relax
1343 }

```

`\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。

```

1344 \def\maketitle{%
1345 \newpage\null
1346 \vskip 2em
1347 \begin{center}%
1348 \let\footnote\thanks
1349 {\LARGE \@title \par}%
1350 \vskip 1.5em
1351 {\large
1352 \lineskip .5em
1353 \begin{tabular}[t]{c}%
1354 \@author
1355 \end{tabular}\par}%
1356 \vskip 1em
1357 {\large \@date}%
1358 \end{center}%
1359 \par\vskip 1.5em
1360 (article | report | kiyou) \ifvoid\@abstractbox\else\centerline{\box\@abstractbox}\vskip1.5em\fi

```

```

1361 }
1362 \fi
1363 </article | book | report | kiyou>
1364 <*jspf>
1365 \newcommand{\maketitle}{\par
1366   \begin{group
1367     \renewcommand\thefootnote{\@fnsymbol\c@footnote}%
1368     \def\@makefnmark{\rlap{\@textsuperscript{\normalfont\@thefnmark}}}%
1369     \long\def\@makefntext##1{\advance\leftskip 3zw
1370       \parindent 1zw\noindent
1371       \llap{\@textsuperscript{\normalfont\@thefnmark}\hskip0.3zw}##1}%
1372     \twocolumn[\@maketitle]%
1373     \plainifnotempty
1374     \@thanks
1375   \endgroup
1376   \setcounter{footnote}{0}%
1377   \global\let\thanks\relax
1378   \global\let\maketitle\relax
1379   \global\let\@thanks\@empty
1380   \global\let\@author\@empty
1381   \global\let\@date\@empty
1382 % \global\let\@title\@empty % \@title は柱に使う
1383   \global\let\title\relax
1384   \global\let\author\relax
1385   \global\let\date\relax
1386   \global\let\and\relax
1387   \ifx\authors@mail\@undefined\else{%
1388     \def\@makefntext{\advance\leftskip 3zw \parindent -3zw}%
1389     \footnotetext[0]{\itshape\authors@mail}%
1390   }\fi
1391   \global\let\authors@mail\@undefined}
1392 \def\@maketitle{%
1393   \newpage\null
1394   \vskip 6em % used to be 2em
1395   \begin{center}
1396     \let\footnote\thanks
1397     \ifx\@title\@undefined\else{\LARGE\headfont\@title\par}\fi
1398     \lineskip .5em
1399     \ifx\@author\@undefined\else
1400       \vskip 1em
1401       \begin{tabular}[t]{c}%
1402         \@author
1403       \end{tabular}\par
1404     \fi
1405     \ifx\@etitle\@undefined\else
1406       \vskip 1em
1407       {\large \@etitle \par}%
1408     \fi
1409     \ifx\@eauthor\@undefined\else

```

```

1410     \vskip 1em
1411     \begin{tabular}[t]{c}%
1412         \@eauthor
1413     \end{tabular}\par
1414     \fi
1415     \vskip 1em
1416     \@date
1417 \end{center}
1418 \vskip 1.5em
1419 \centerline{\box\@abstractbox}
1420 \ifx\@keywords\@undefined\else
1421     \vskip 1.5em
1422     \centerline{\parbox{157\jsc@mmm}{\textsf{Keywords:}}\ \small\@keywords}}
1423 \fi
1424 \vskip 1.5em}
1425 </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 `\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。

```

1426 \def\@startsection#1#2#3#4#5#6{%

```

```

1427 \if@noskipsec \leavevmode \fi
1428 \par
1429 % 見出し上の空きを \@tempkipa にセットする
1430 \@tempkipa #4\relax
1431 % \@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
1432 \if@english \@afterindentfalse \else \@afterindenttrue \fi
1433 % 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
1434 \ifdim \@tempkipa <\z@
1435   \@tempkipa -\@tempkipa \@afterindentfalse
1436 \fi
1437 \if@nobreak
1438 % \everypar{\everyparhook}% これは間違い
1439 \everypar{}%
1440 \else
1441 \addpenalty\@secpenalty
1442 % 次の行は削除
1443 % \addvspace\@tempkipa
1444 % 次の \noindent まで追加
1445 \ifdim \@tempkipa >\z@
1446 \if@slide\else
1447 \null
1448 \vspace*{-\baselineskip}%
1449 \fi
1450 \vskip\@tempkipa
1451 \fi
1452 \fi
1453 \noindent
1454 % 追加終わり
1455 \@ifstar
1456 {\@ssect{#3}{#4}{#5}{#6}}%
1457 {\@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。

```

1458 \def\@sect#1#2#3#4#5#6[#7]#8{%
1459 \ifnum #2>\c@secnumdepth
1460 \let\@svsec\@empty
1461 \else
1462 \refstepcounter{#1}%
1463 \protected@edef\@svsec{\@secntformat{#1}\relax}%
1464 \fi
1465 % 見出し後の空きを \@tempkipa にセット
1466 \@tempkipa #5\relax
1467 % 条件判断の順序を入れ替えました
1468 \ifdim \@tempkipa<\z@
1469 \def\@svsechd{%
1470 #6{\hskip #3\relax
1471 \@svsec #8}%
1472 \csname #1mark\endcsname{#7}}%

```

```

1473 \addcontentsline{toc}{#1}{%
1474 \ifnum #2>\c@secnumdepth \else
1475 \protect\numberline{\csname the#1\endcsname}%
1476 \fi
1477 #7}}% 目次にフルネームを載せるなら #8
1478 \else
1479 \begingroup
1480 \interlinepenalty \@M % 下から移動
1481 #6{%
1482 \@hangfrom{\hskip #3\relax\@svsec}%
1483 % \interlinepenalty \@M % 上に移動
1484 #8\@@par}%
1485 \endgroup
1486 \csname #1mark\endcsname{#7}%
1487 \addcontentsline{toc}{#1}{%
1488 \ifnum #2>\c@secnumdepth \else
1489 \protect\numberline{\csname the#1\endcsname}%
1490 \fi
1491 #7}}% 目次にフルネームを載せるならここは #8
1492 \fi
1493 \@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。

```

1494 \def\@xsect#1{%
1495 % 見出しの後ろの空きを \@tempskipa にセット
1496 \@tempskipa #1\relax
1497 % 条件判断の順序を変えました
1498 \ifdim \@tempskipa<\z@
1499 \nobreakfalse
1500 \global\@noskipsectrue
1501 \everypar{%
1502 \if@noskipsec
1503 \global\@noskipsecfalse
1504 {\setbox\z@\lastbox}%
1505 \clubpenalty\@M
1506 \begingroup \@svsechd \endgroup
1507 \unskip
1508 \@tempskipa #1\relax
1509 \hskip -\@tempskipa
1510 \else
1511 \clubpenalty \@clubpenalty
1512 \everypar{\everyparhook}%
1513 \fi\everyparhook}%
1514 \else
1515 \par \nobreak

```



```

1516 \vskip \@tempskipa
1517 \@afterheading
1518 \fi
1519 \if@slide
1520 {\vskip\if@twocolumn-5\jsc@empt\else-6\jsc@empt\fi
1521 \maybeblue\hrule height0\jsc@mpt depth1\jsc@mpt
1522 \vskip\if@twocolumn 4\jsc@empt\else 7\jsc@empt\fi\relax}%
1523 \fi
1524 \par % 2000-12-18
1525 \ignorespaces}
1526 \def\@ssect#1#2#3#4#5{%
1527 \@tempskipa #3\relax
1528 \ifdim \@tempskipa<\z@
1529 \def\@svsechd{#4{\hskip #1\relax #5}}%
1530 \else
1531 \begingroup
1532 #4{%
1533 \@hangfrom{\hskip #1}%
1534 \interlinepenalty \@M #5\@@par}%
1535 \endgroup
1536 \fi
1537 \@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 1538 \newcommand*\chaptermark[1]{}
\subsubsectionmark 1539 % \newcommand*\sectionmark[1]{}
\paragraphmark 1540 % \newcommand*\subsectionmark[1]{}
\subparagraphmark 1541 % \newcommand*\subsubsectionmark[1]{}
1542 % \newcommand*\paragraphmark[1]{}
1543 % \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。

```

1544 (!book&!report)\setcounter{secnumdepth}{3}
1545 (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 1546 \newcounter{part}
\c@subsubsection 1547 (book | report)\newcounter{chapter}
1548 (book | report)\newcounter{section}[chapter]
\c@paragraph 1549 (!book&!report)\newcounter{section}
\c@subparagraph 1550 \newcounter{subsection}[section]
1551 \newcounter{subsubsection}[subsection]

```

```
1552 \newcounter{paragraph}[subsubsection]
1553 \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。
\thesection      \arabic{COUNTER}    1, 2, 3, ...
\thesubsection   \roman{COUNTER}     i, ii, iii, ...
\thesubsubsection \Roman{COUNTER}    I, II, III, ...
\theparagraph    \alph{COUNTER}      a, b, c, ...
\thesubparagraph \Alph{COUNTER}      A, B, C, ...
                \kansuji{COUNTER}   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。

```
1554 \renewcommand{\thepart}{\@Roman\c@part}
1555 (!book&!report)% \renewcommand{\thesection}{\@arabic\c@section}
1556 (!book&!report)\renewcommand{\thesection}{\presectionname\@arabic\c@section\postsectionname}
1557 (!book&!report)\renewcommand{\thesubsection}{\@arabic\c@section.\@arabic\c@subsection}
1558 (*book | report)
1559 \renewcommand{\thechapter}{\@arabic\c@chapter}
1560 \renewcommand{\thesection}{\thechapter.\@arabic\c@section}
1561 \renewcommand{\thesubsection}{\thesection.\@arabic\c@subsection}
1562 </book | report)
1563 \renewcommand{\thesubsubsection}{%
1564   \thesubsection.\@arabic\c@subsubsection}
1565 \renewcommand{\theparagraph}{%
1566   \thesubsubsection.\@arabic\c@paragraph}
1567 \renewcommand{\thesubparagraph}{%
1568   \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。
```

```
1569 (book | report)\newcommand{\@chapapp}{\prechaptername}
1570 (book | report)\newcommand{\@chappos}{\postchaptername}
```

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

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

```

1571 (*book)
1572 \newcommand\frontmatter{%
1573   \pltx@cleartooddpage
1574   \@mainmatterfalse
1575   \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。

```

1576 \newcommand\mainmatter{%
1577   \pltx@cleartooddpage
1578   \@mainmattertrue
1579   \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。

```

1580 \newcommand\backmatter{%
1581   \if@openleft
1582     \cleardoublepage
1583   \else\if@openright
1584     \cleardoublepage
1585   \else
1586     \clearpage
1587   \fi\fi
1588   \@mainmatterfalse}
1589 (/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。

```

1590 (*!book&!report)
1591 \newcommand\part{%
1592   \if@noskipsec \leavevmode \fi
1593   \par
1594   \addvspace{4ex}%
1595   \if@english \afterindentfalse \else \@afterindenttrue \fi
1596   \secdef\@part\@spart}
1597 (/!book&!report)

```

book および report クラスの場合は、少し複雑です。

```
1598 (*book | report)
1599 \newcommand\part{%
1600   \if@openleft
1601     \cleardoublepage
1602   \else\if@openright
1603     \cleardoublepage
1604   \else
1605     \clearpage
1606   \fi\fi
1607   \thispagestyle{empty}% 欧文用標準スタイルでは plain
1608   \if@twocolumn
1609     \onecolumn
1610     \@restonecoltrue
1611   \else
1612     \@restonecolfalse
1613   \fi
1614   \null\vfil
1615   \secdef\@part\@spart}
1616 \end{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。

```
1617 (*!book&!report)
1618 \def\@part[#1]#2{%
1619   \ifnum \c@secnumdepth >\m@ne
1620     \refstepcounter{part}%
1621     \addcontentsline{toc}{part}{%
1622       \prepartname\thepart\postpartname\hspace{1zw}#1}%
1623   \else
1624     \addcontentsline{toc}{part}{#1}%
1625   \fi
1626   \markboth{}{}%
1627   {\parindent\z@
1628     \raggedright
1629     \interlinepenalty \@M
1630     \normalfont
1631     \ifnum \c@secnumdepth >\m@ne
1632       \Large\headfont\prepartname\thepart\postpartname
1633       \par\nobreak
1634     \fi
1635     \huge \headfont #2%
1636     \markboth{}{}\par}%
1637   \nobreak
1638   \vskip 3ex
1639   \@afterheading}
1640 \end{!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。

```
1641 (*book | report)
1642 \def\@part[#1]#2{%
1643   \ifnum \c@secnumdepth >-2\relax
1644     \refstepcounter{part}%
1645     \addcontentsline{toc}{part}{%
1646       \prepartname\thepart\postpartname\hspace{1zw}#1}%
1647   \else
1648     \addcontentsline{toc}{part}{#1}%
1649   \fi
1650   \markboth{}{}%
1651   {\centering
1652     \interlinepenalty \@M
1653     \normalfont
1654     \ifnum \c@secnumdepth >-2\relax
1655       \huge\headfont \prepartname\thepart\postpartname
1656       \par\vskip20\jsc@empt
1657     \fi
1658     \Huge \headfont #2\par}%
1659   \@endpart}
1660 </book | report)
```

\@spart 番号を付けない部です。

```
1661 (*!book&!report)
1662 \def\@spart#1{%
1663   \parindent \z@ \raggedright
1664   \interlinepenalty \@M
1665   \normalfont
1666   \huge \headfont #1\par}%
1667 \nobreak
1668 \vskip 3ex
1669 \@afterheading}
1670 </!book&!report)
1671 (*book | report)
1672 \def\@spart#1{%
1673   \centering
1674   \interlinepenalty \@M
1675   \normalfont
1676   \Huge \headfont #1\par}%
1677 \@endpart}
1678 </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。

```
1679 (*book | report)
```

```

1680 \def\@endpart{\vfil\newpage
1681   \if@twoside
1682   \if@openleft %% added (2017/02/24)
1683     \null\thispagestyle{empty}\newpage
1684   \else\if@openright %% added (2016/12/13)
1685     \null\thispagestyle{empty}\newpage
1686   \fi\fi %% added (2016/12/13, 2017/02/24)
1687   \fi
1688   \if@restonecol
1689     \twocolumn
1690   \fi}
1691 </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。

```

1692 (*book | report)
1693 \newcommand{\chapter}{%
1694   \if@openleft\cleardoublepage\else
1695   \if@openright\cleardoublepage\else\clearpage\fi\fi
1696   \plainifnotempty % 元: \thispagestyle{plain}
1697   \global\@topnum\z@
1698   \if@english \@afterindentfalse \else \@afterindenttrue \fi
1699   \secdef
1700     {\@omit@numberfalse\@chapter}%
1701     {\@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。

```

1702 \def\@chapter[#1]#2{%
1703   \ifnum \c@secnumdepth >\m@ne
1704 (book)   \if@mainmatter
1705           \refstepcounter{chapter}%
1706           \typeout{\@chapapp\thechapter\@chappos}%
1707           \addcontentsline{toc}{chapter}{%
1708             {\protect\numberline
1709              % {\if@english\thechapter\else\@chapapp\thechapter\@chappos\fi}}%
1710             {\@chapapp\thechapter\@chappos}%
1711             #1}%
1712 (book)   \else\addcontentsline{toc}{chapter}{#1}\fi
1713   \else
1714     \addcontentsline{toc}{chapter}{#1}%
1715   \fi
1716   \chaptermark{#1}%
1717   \addtocontents{lof}{\protect\addvspace{10\jsc@mpt}}%
1718   \addtocontents{lot}{\protect\addvspace{10\jsc@mpt}}%
1719   \if@twocolumn

```

```

1720 \topnewpage[\@makechapterhead{#2}]%
1721 \else
1722 \@makechapterhead{#2}%
1723 \@afterheading
1724 \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。

```

1725 \def\@makechapterhead#1{%
1726 \vspace*{2\Cvs}% 欧文は 50pt
1727 {\parindent \z@ \raggedright \normalfont
1728 \ifnum \c@secnumdepth >\m@ne
1729 (book) \if@mainmatter
1730 \huge\headfont \@chapapp\thechapter\@chappos
1731 \par\nobreak
1732 \vskip \Cvs % 欧文は 20pt
1733 (book) \fi
1734 \fi
1735 \interlinepenalty\@M
1736 \Huge \headfont #1\par\nobreak
1737 \vskip 3\Cvs}} % 欧文は 40pt

```

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

```

1738 \def\@chapter#1{%
1739 \chaptermark{#1}%
1740 \if@twocolumn
1741 \topnewpage[\@makeschapterhead{#1}]%
1742 \else
1743 \@makeschapterhead{#1}\@afterheading
1744 \fi}

```

`\@makeschapterhead` 番号なしの章見出しです。

```

1745 \def\@makeschapterhead#1{%
1746 \vspace*{2\Cvs}% 欧文は 50pt
1747 {\parindent \z@ \raggedright
1748 \normalfont
1749 \interlinepenalty\@M
1750 \Huge \headfont #1\par\nobreak
1751 \vskip 3\Cvs}} % 欧文は 40pt
1752 </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。

```

1753 \if@twocolumn
1754 \newcommand{\section}{%
1755 (jspf)\ifx\maketitle\relax\else\maketitle\fi

```

```

1756 \startsection{section}{1}{\z0}%
1757 (!kiyou) {0.6\Cvs}{0.4\Cvs}%
1758 (kiyou) {\Cvs}{0.5\Cvs}%
1759 % {\normalfont\large\headfont\@secapp}}
1760 {\normalfont\large\headfont\raggedright}}
1761 \else
1762 \newcommand{\section}{%
1763 \if@slide\clearpage\fi
1764 \startsection{section}{1}{\z0}%
1765 {\Cvs \@plus.5\Cdp \@minus.2\Cdp}% 前アキ
1766 {.5\Cvs \@plus.3\Cdp}% 後アキ
1767 % {\normalfont\Large\headfont\@secapp}}
1768 {\normalfont\Large\headfont\raggedright}}
1769 \fi

```

`\subsection` 同上です。

```

1770 \if@twocolumn
1771 \newcommand{\subsection}{\startsection{subsection}{2}{\z0}%
1772 {\z0}{\if@slide .4\Cvs \else \z0 \fi}%
1773 {\normalfont\normalsize\headfont}}
1774 \else
1775 \newcommand{\subsection}{\startsection{subsection}{2}{\z0}%
1776 {\Cvs \@plus.5\Cdp \@minus.2\Cdp}% 前アキ
1777 {.5\Cvs \@plus.3\Cdp}% 後アキ
1778 {\normalfont\large\headfont}}
1779 \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)。

```

1780 \if@twocolumn
1781 \newcommand{\subsubsection}{\startsection{subsubsection}{3}{\z0}%
1782 {\z0}{\if@slide .4\Cvs \else \z0 \fi}%
1783 {\normalfont\normalsize\headfont}}
1784 \else
1785 \newcommand{\subsubsection}{\startsection{subsubsection}{3}{\z0}%
1786 {\Cvs \@plus.5\Cdp \@minus.2\Cdp}%
1787 {\if@slide .5\Cvs \@plus.3\Cdp \else \z0 \fi}%
1788 {\normalfont\normalsize\headfont}}
1789 \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。


```

1790 <!jspf>\newcommand{\jsParagraphMark}{■}
1791 \if@twocolumn
1792   \newcommand{\paragraph}{\@startsection{paragraph}{4}{\z0}%
1793     {\z0}{\if@slide .4\Cvs \else -1zw\fi}% 改行せず 1zw のアキ
1794 (jspf)   {\normalfont\normalsize\headfont}}
1795 <!jspf>   {\normalfont\normalsize\headfont\jsParagraphMark}}
1796 \else
1797   \newcommand{\paragraph}{\@startsection{paragraph}{4}{\z0}%
1798     {0.5\Cvs \@plus.5\Cdp \@minus.2\Cdp}%
1799     {\if@slide .5\Cvs \@plus.3\Cdp \else -1zw\fi}% 改行せず 1zw のアキ
1800 (jspf)   {\normalfont\normalsize\headfont}}
1801 <!jspf>   {\normalfont\normalsize\headfont\jsParagraphMark}}
1802 \fi

```

`\subparagraph` 見出しの後ろで改行されません。

```

1803 \if@twocolumn
1804   \newcommand{\subparagraph}{\@startsection{subparagraph}{5}{\z0}%
1805     {\z0}{\if@slide .4\Cvs \@plus.3\Cdp \else -1zw\fi}%
1806     {\normalfont\normalsize\headfont}}
1807 \else
1808   \newcommand{\subparagraph}{\@startsection{subparagraph}{5}{\z0}%
1809     {\z0}{\if@slide .5\Cvs \@plus.3\Cdp \else -1zw\fi}%
1810     {\normalfont\normalsize\headfont}}
1811 \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。

```

1812 \if@slide
1813   \setlength\leftmargini{1zw}
1814 \else
1815   \if@twocolumn
1816     \setlength\leftmargini{2zw}
1817   \else
1818     \setlength\leftmargini{3zw}
1819   \fi
1820 \fi

```

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

`\leftmarginiv` ¹⁸²¹ `\if@slide`

`\leftmarginv`

`\leftmarginvi`

```

1822 \setlength\leftmarginii {1zw}
1823 \setlength\leftmarginiii{1zw}
1824 \setlength\leftmarginiv {1zw}
1825 \setlength\leftmarginv {1zw}
1826 \setlength\leftmarginvi {1zw}
1827 \else
1828 \setlength\leftmarginii {2zw}
1829 \setlength\leftmarginiii{2zw}
1830 \setlength\leftmarginiv {2zw}
1831 \setlength\leftmarginv {1zw}
1832 \setlength\leftmarginvi {1zw}
1833 \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。

```

1834 \setlength \labelsep {0.5zw} % .5em
1835 \setlength \labelwidth{\leftmargini}
1836 \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。

```

1837 \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 1838 \@beginparpenalty -\@lowpenalty
\@itempenalty 1839 \@endparpenalty -\@lowpenalty
1840 \@itempenalty -\@lowpenalty

```

`\@listi` `\@listi` は `\leftmargin`, `\parsep`, `\topsep`, `\itemsep` などのトップレベルの定義を `\@listI` します。この定義は、フォントサイズコマンドによって変更されます（たとえば `\small` の 中では小さい値に設定されます）。このため、`\normalsize` がすべてのパラメータを戻せる ように、`\@listI` で `\@listi` のコピーを保存します。元の値はかなり複雑ですが、ここで は簡素化してしまいました。特に最初と最後に行送りの半分の空きが入るようにしてありま す。アスキーの標準スタイルではトップレベルの `itemize`, `enumerate` 環境でだけ最初と 最後に行送りの半分の空きが入るようになっていました。

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

```

1841 \def\@listi{\leftmargin\leftmargini
1842 \parsep z@
1843 \topsep 0.5\baselineskip
1844 \itemsep z@ \relax}
1845 \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）。

```

1846 \@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 1847 \def\@listii{\leftmargin\leftmarginii
\@listiv
\@listv
\@listvi

```

```

1848 \labelwidth\leftmarginii \advance\labelwidth-\labelsep
1849 \topsep \z@
1850 \parsep \z@
1851 \itemsep\parsep}
1852 \def\@listiii{\leftmargin\leftmarginiii
1853 \labelwidth\leftmarginiii \advance\labelwidth-\labelsep
1854 \topsep \z@
1855 \parsep \z@
1856 \itemsep\parsep}
1857 \def\@listiv {\leftmargin\leftmarginiv
1858 \labelwidth\leftmarginiv
1859 \advance\labelwidth-\labelsep}
1860 \def\@listv {\leftmargin\leftmarginv
1861 \labelwidth\leftmarginv
1862 \advance\labelwidth-\labelsep}
1863 \def\@listvi {\leftmargin\leftmarginvi
1864 \labelwidth\leftmarginvi
1865 \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。

```

1866 \renewcommand{\theenumi}{\@arabic\c@enumi}
1867 \renewcommand{\theenumii}{\@alph\c@enumii}
1868 \renewcommand{\theenumiii}{\@roman\c@enumiii}
1869 \renewcommand{\theenumiv}{\@Alph\c@enumiv}

```

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

```

\labelenumiv 1870 \newcommand{\labelenumi}{\theenumi.}
1871 \newcommand{\labelenumii}{\inhibitglue (\theenumii) \inhibitglue}
1872 \newcommand{\labelenumiii}{\theenumiii.}
1873 \newcommand{\labelenumiv}{\theenumiv.}

```

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

```

\p@enumiv 1874 \renewcommand{\p@enumii}{\theenumi}
1875 \renewcommand{\p@enumiii}{\theenumi\inhibitglue (\theenumii) }
1876 \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 1877 \newcommand\labelitemii{\textbullet}
\labelitemiii 1878 \newcommand\labelitemiii{\normalfont\bfseries \textendash}
\labelitemiv 1879 \newcommand\labelitemiv{\textasteriskcentered}
\labelitemv 1880 \newcommand\labelitemv{\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してしまいます。これを解決した新しい `description` の実装です。

```
1881 \newenvironment{description}{%
1882   \list{}{%
1883     \labelwidth=\leftmargin
1884     \labelsep=1zw
1885     \advance \labelwidth by -\labelsep
1886     \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。

```
1887 \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。

```
1888 (*book)
1889 \newenvironment{abstract}{%
1890   \begin{list}{}{%
1891     \listparindent=1zw
1892     \itemindent=\listparindent
1893     \rightmargin=0pt
1894     \leftmargin=5zw}\item[]}{\end{list}\vspace{\baselineskip}}
1895 </book>
1896 (*article | report | kiyou)
1897 \newbox\@abstractbox
1898 \if@titlepage
1899   \newenvironment{abstract}{%
1900     \titlepage
1901     \null\vfil
1902     \@beginparpenalty\@lowpenalty
1903     \begin{center}%
1904       \headfont \abstractname
1905       \@endparpenalty\@M
1906     \end{center}}%
```

```

1907  {\par\vfil\null\endtitlepage}
1908 \else
1909  \newenvironment{abstract}{%
1910    \if@twocolumn
1911      \ifx\maketitle\relax
1912        \section*{\abstractname}%
1913      \else
1914        \global\setbox\@abstractbox\hbox\bgroup
1915        \begin{minipage}[b]{\textwidth}
1916          \small\parindent1zw
1917          \begin{center}%
1918            {\headfont \abstractname\vspace{-.5em}\vspace{\z@}}%
1919          \end{center}%
1920          \list{}{%
1921            \listparindent\parindent
1922            \itemindent \listparindent
1923            \rightmargin \leftmargin}%
1924          \item\relax
1925        \fi
1926      \else
1927        \small
1928        \begin{center}%
1929          {\headfont \abstractname\vspace{-.5em}\vspace{\z@}}%
1930        \end{center}%
1931        \list{}{%
1932          \listparindent\parindent
1933          \itemindent \listparindent
1934          \rightmargin \leftmargin}%
1935        \item\relax
1936      \fi}{\if@twocolumn
1937        \ifx\maketitle\relax
1938        \else
1939          \endlist\end{minipage}\egroup
1940        \fi
1941      \else
1942        \endlist
1943      \fi}
1944 \fi
1945 </article | report | kiyou>
1946 <*jspf>
1947 \newbox\@abstractbox
1948 \newenvironment{abstract}{%
1949   \global\setbox\@abstractbox\hbox\bgroup
1950   \begin{minipage}[b]{157\jsc@mmm}{\sffamily Abstract}\par
1951     \small
1952     \if@english \parindent6\jsc@mmm \else \parindent1zw \fi}%
1953   {\end{minipage}\egroup}
1954 </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。

```
1955 <*jspf>
1956 %\newbox\@keywordsbox
1957 %\newenvironment{keywords}{%
1958 % \global\setbox\@keywordsbox\hbox\bgroup
1959 % \begin{minipage}[b]{1570\jsc@mmm}{\sffamily Keywords:}\par
1960 % \small\parindent0zw}%
1961 % {\end{minipage}\egroup}
1962 </jspf>
```

■verse 環境

`verse` 詩のための `verse` 環境です。

```
1963 \newenvironment{verse}{%
1964 \let \\\=@centercr
1965 \list{}{%
1966 \itemsep \z@
1967 \itemindent -2zw % 元: -1.5em
1968 \listparindent\itemindent
1969 \rightmargin \z@
1970 \advance\leftmargin 2zw}% 元: 1.5em
1971 \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。

```
1972 \newenvironment{quotation}{%
1973 \list{}{%
1974 \listparindent\parindent
1975 \itemindent\listparindent
1976 \rightmargin \z@}%
1977 \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。

```
1978 \newenvironment{quote}%
1979 {\list{}{\rightmargin\z@}\item\relax}{\endlist}
```

■定理など `ltthm.dtx` 参照。たとえば次のように定義します。

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

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

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

```
1980 \def\@begintheorem#1#2{\trivlist\labelsep=1zw
1981   \item[\hskip \labelsep{\headfont #1\ #2}]}
1982 \def\@opargbegintheorem#1#2#3{\trivlist\labelsep=1zw
1983   \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 p_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 p_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。

```
1984 \newenvironment{titlepage}{%
1985 (book)   \pltx@cleartooddpage %% 2017-02-24
1986   \if@twocolumn
1987     \@restonecoltrue\onecolumn
1988   \else
1989     \@restonecolfalse\newpage
1990   \fi
1991   \thispagestyle{empty}%
1992   \ifodd\c@page\setcounter{page}\@ne\else\setcounter{page}\z@\fi %% 2017-02-24
1993 }%
1994 {\if@restonecol\twocolumn \else \newpage \fi
1995   \if@twoside\else
1996     \setcounter{page}\@ne
1997   \fi}
```

■付録

`\appendix` 本文と付録を分離するコマンドです。

```
1998 (*!book&!report)
1999 \newcommand{\appendix}{\par
2000   \setcounter{section}{0}%
2001   \setcounter{subsection}{0}%
2002   \gdef\presectionname{\appendixname}%
2003   \gdef\postsectionname{}}%
2004 % \gdef\thesection{\@Alph\c@section}% [2003-03-02]
2005 \gdef\thesection{\presectionname\@Alph\c@section\postsectionname}%
2006 \gdef\thesubsection{\@Alph\c@section.\@arabic\c@subsection}}
2007 (!book&!report)
2008 (*book | report)
2009 \newcommand{\appendix}{\par
```

```

2010 \setcounter{chapter}{0}%
2011 \setcounter{section}{0}%
2012 \gdef\@chapapp{\appendixname}%
2013 \gdef\@chappos{}%
2014 \gdef\thechapter{\@Alph@c@chapter}
2015 </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。

```
2016 \setlength\arraycolsep{5\jsc@empt}
```

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

```
2017 \setlength\tabcolsep{6\jsc@empt}
```

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

```
2018 \setlength\arrayrulewidth{.4\jsc@empt}
```

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

```
2019 \setlength\doublerulesep{2\jsc@empt}
```

■tabbing 環境

`\tabbingsep` \’ コマンドで入るアキです。

```
2020 \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。

```
2021 \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。

```
2022 \setlength\fboxsep{3\jsc@empt}
```

```
2023 \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。

```
2024 (!book&!report)\renewcommand \theequation {\@arabic@c@equation}
```



```

2025 (*book | report)
2026 \@addtoreset{equation}{chapter}
2027 \renewcommand\theequation
2028   {\ifnum \c@chapter>\z@ \thechapter.\fi \@arabic\c@equation}
2029 \end{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。

```
2030 % \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。

```
2031 % \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。

```
2032 % \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。

```

2033 (*!book&!report)
2034 \newcounter{figure}
2035 \renewcommand \thefigure {\@arabic\c@figure}
2036 \end{!book&!report}
2037 (*book | report)
2038 \newcounter{figure}[chapter]
2039 \renewcommand \thefigure
2040   {\ifnum \c@chapter>\z@ \thechapter.\fi \@arabic\c@figure}
2041 \end{book | report}

```

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

`\ext@figure`

`\fnum@figure`

```

2042 \def\fps@figure{tbp}
2043 \def\ftype@figure{1}
2044 \def\ext@figure{lof}
2045 \def\fnun@figure{\figurename\nobreak\thefigure}

```

`figure` * 形式は段抜きフロートです。

```

figure* 2046 \newenvironment{figure}%
2047         {\@float{figure}}%
2048         {\end@float}
2049 \newenvironment{figure*}%
2050         {\@dblfloat{figure}}%
2051         {\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。

```

2052 <!*book&!report)
2053 \newcounter{table}
2054 \renewcommand\thetable{\@arabic\c@table}
2055 <!/book&!report)
2056 (*book | report)
2057 \newcounter{table}[chapter]
2058 \renewcommand \thetable
2059     {\ifnum \c@chapter>\z@ \thechapter.\fi \@arabic\c@table}
2060 </book | report)

```

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

```

\ext@table 2061 \def\fps@table{tbp}
2062 \def\ftype@table{2}
\fnun@table 2063 \def\ext@table{lot}
2064 \def\fnun@table{\tablename\nobreak\thetable}

```

`table` * は段抜きフロートです。

```

table* 2065 \newenvironment{table}%
2066         {\@float{table}}%
2067         {\end@float}
2068 \newenvironment{table*}%
2069         {\@dblfloat{table}}%
2070         {\end@dblfloat}

```

8.6 キャプション

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

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

```
2071 \newlength\abovecaptionskip
2072 \newlength\belowcaptionskip
2073 \setlength\abovecaptionskip{5\jsc@mp} % 元: 10\p@
2074 \setlength\belowcaptionskip{5\jsc@mp} % 元: 0\p@
```

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

[2003-11-05] ロジックを少し変えてみました。

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

```
2075 (*!jspf)
2076 % \long\def\@makecaption#1#2{\small
2077 %   \advance\leftskip10\jsc@mmm
2078 %   \advance\rightskip10\jsc@mmm
2079 %   \vskip\abovecaptionskip
2080 %   \sbox\@tempboxa{#1\hskip1zw\relax #2}%
2081 %   \ifdim \wd\@tempboxa >\hsize
2082 %     #1\hskip1zw\relax #2\par
2083 %   \else
2084 %     \global \@minipagefalse
2085 %     \hb@xt@\hsize{\hfil\box\@tempboxa\hfil}%
2086 %   \fi
2087 %   \vskip\belowcaptionskip}}
2088 \long\def\@makecaption#1#2{\small
2089   \advance\leftskip .0628\linewidth
2090   \advance\rightskip .0628\linewidth
2091   \vskip\abovecaptionskip
2092   \sbox\@tempboxa{#1\hskip1zw\relax #2}%
2093   \ifdim \wd\@tempboxa <\hsize \centering \fi
2094   #1{\hskip1zw\relax}#2\par
2095   \vskip\belowcaptionskip}}
2096 </!jspf)
2097 (*jspf)
2098 \long\def\@makecaption#1#2{%
2099   \vskip\abovecaptionskip
2100   \sbox\@tempboxa{\small\sffamily #1\quad #2}%
2101   \ifdim \wd\@tempboxa >\hsize
2102     {\small\sffamily
2103       \list{#1}{%
2104         \renewcommand{\makelabel}[1]{##1\hfil}
2105         \itemsep \z@
2106         \itemindent \z@
2107         \labelsep \z@
2108         \labelwidth 11\jsc@mmm
```

```

2109     \listparindent\z@
2110     \leftmargin 11\jsc@mmm}\item\relax #2\endlist}
2111 \else
2112   \global \@minipagefalse
2113   \hb@xt@\hsize{\hfil\box\@tempboxa\hfil}%
2114   \fi
2115   \vskip\belowcaptionskip}
2116 \</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 2117 \DeclareOldFontCommand{\mc}{\normalfont\mcfamily}{\mathmc}
\rm 2118 \DeclareOldFontCommand{\gt}{\normalfont\gtfamily}{\mathgt}
2119 \DeclareOldFontCommand{\rm}{\normalfont\rmfamily}{\mathrm}
\sf 2120 \DeclareOldFontCommand{\sf}{\normalfont\sffamily}{\mathsf}
\tt 2121 \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。

```
2122 \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ありません (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。

```

2123 \DeclareOldFontCommand{\it}{\normalfont\itshape}{\mathit}
2124 \DeclareOldFontCommand{\sl}{\normalfont\slshape}{\@nomath\sl}
2125 \DeclareOldFontCommand{\sc}{\normalfont\scshape}{\@nomath\sc}

```

`\cal` 数式モード以外では何もありません (警告を出します)。

```

\mit 2126 \DeclareRobustCommand*\cal{\@fontswitch\relax\mathcal}
2127 \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。

```
2128 \newcommand\@pnumwidth{1.55em}
2129 \newcommand\@tocrmarg{2.55em}
2130 \newcommand\@dotsep{4.5}
2131 (!book&!report)\setcounter{tocdepth}{2}
2132 (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)

```
2133 \newdimen\jsc@tocl@width
2134 \newcommand{\tableofcontents}{%
2135 (*book | report)
2136 \settowidth\jsc@tocl@width{\headfont\prechaptername\postchaptername}%
2137 \settowidth\@tempdima{\headfont\appendixname}%
2138 \ifdim\jsc@tocl@width<\@tempdima \setlength\jsc@tocl@width{\@tempdima}\fi
2139 \ifdim\jsc@tocl@width<2zw \divide\jsc@tocl@width by 2 \advance\jsc@tocl@width 1zw\fi
```

```

2140 \if@twocolumn
2141   \@restonecoltrue\onecolumn
2142 \else
2143   \@restonecolfalse
2144 \fi
2145 \chapter*{\contentsname}%
2146 \@mkboth{\contentsname}{}%
2147 </book | report>
2148 <!*book&!report>
2149 \settowidth\jsc@tocl@width{\headfont\presectionname\postsectionname}%
2150 \settowidth\@tempdima{\headfont\appendixname}%
2151 \ifdim\jsc@tocl@width<\@tempdima\relax\setlength\jsc@tocl@width{\@tempdima}\fi
2152 \ifdim\jsc@tocl@width<2zw \divide\jsc@tocl@width by 2 \advance\jsc@tocl@width 1zw\fi
2153 \section*{\contentsname}%
2154 \@mkboth{\contentsname}{\contentsname}%
2155 </!book&!report>
2156 \@starttoc{toc}%
2157 <book | report> \if@restonecol\twocolumn\fi
2158 }

```

\l@part 部の目次です。

```

2159 \newcommand*\l@part}[2]{%
2160   \ifnum \c@tocdepth >-2\relax
2161   <!book&!report> \addpenalty\@secpenalty
2162   <book | report> \addpenalty{-\@highpenalty}%
2163   \addvspace{2.25em \@plus\jsc@mpt}%
2164   \begingroup
2165     \parindent \z@
2166   % \@pnumwidth should be \@tocrmarg
2167   % \rightskip \@pnumwidth
2168   \rightskip \@tocrmarg
2169   \parfillskip -\rightskip
2170   {\leavevmode
2171     \large \headfont
2172     \setlength\@lnumwidth{4zw}%
2173     #1\hfil \hb@xt@\@pnumwidth{\hss #2}}\par
2174   \nobreak
2175   <book | report> \global\@nobreaktrue
2176   <book | report> \everypar{\global\@nobreakfalse\everypar{}}%
2177   \endgroup
2178   \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)

```

2179 <*book | report>
2180 \newcommand*\l@chapter}[2]{%
2181   \ifnum \c@tocdepth >\m@ne
2182     \addpenalty{-\@highpenalty}%

```

```

2183 \advspace{1.0em \@plus\jsc@empt}
2184 % \vskip 1.0em \@plus\p@ % book.cls では↑がこうなっている
2185 \begingroup
2186 \parindent\z@
2187 % \rightskip\@pnumwidth
2188 \rightskip\@tocrmarg
2189 \parfillskip-\rightskip
2190 \leavevmode\headfont
2191 % \if@english\setlength\@lnumwidth{5.5em}\else\setlength\@lnumwidth{4.683zw}\fi
2192 \setlength\@lnumwidth{\jsc@tocl@width}\advance\@lnumwidth 2.683zw
2193 \advance\leftskip\@lnumwidth \hskip-\leftskip
2194 #1\nobreak\hfil\nobreak\hbox to\@pnumwidth{\hss#2}\par
2195 \penalty\@highpenalty
2196 \endgroup
2197 \fi}
2198 </book | report)

```

`\l@section` 節の目次です。

```

2199 (*!book&!report)
2200 \newcommand*{\l@section}[2]{%
2201 \ifnum \c@tocdepth >\z@
2202 \addpenalty{\@secpenalty}%
2203 \advspace{1.0em \@plus\jsc@empt}%
2204 \begingroup
2205 \parindent\z@
2206 % \rightskip\@pnumwidth
2207 \rightskip\@tocrmarg
2208 \parfillskip-\rightskip
2209 \leavevmode\headfont
2210 %\setlength\@lnumwidth{4zw}% 元 1.5em [2003-03-02]
2211 \setlength\@lnumwidth{\jsc@tocl@width}\advance\@lnumwidth 2zw
2212 \advance\leftskip\@lnumwidth \hskip-\leftskip
2213 #1\nobreak\hfil\nobreak\hbox to\@pnumwidth{\hss#2}\par
2214 \endgroup
2215 \fi}
2216 </!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。

```

2217 <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] ここも `\jsc@tocl@width` から決めるようにしてみました。(by ts)

```

\l@subparagraph 2218 (*!book&!report)
2219 % \newcommand*{\l@subsection} {\@dottedtocline{2}{1.5em}{2.3em}}
2220 % \newcommand*{\l@subsubsection}{\@dottedtocline{3}{3.8em}{3.2em}}
2221 % \newcommand*{\l@paragraph} {\@dottedtocline{4}{7.0em}{4.1em}}

```

```

2222 % \newcommand*\l@subparagraph {\@dottedtocline{5}{10em}{5em}}
2223 %
2224 % \newcommand*\l@subsection {\@dottedtocline{2}{1zw}{3zw}}
2225 % \newcommand*\l@subsubsection{\@dottedtocline{3}{2zw}{3zw}}
2226 % \newcommand*\l@paragraph {\@dottedtocline{4}{3zw}{3zw}}
2227 % \newcommand*\l@subparagraph {\@dottedtocline{5}{4zw}{3zw}}
2228 %
2229 \newcommand*\l@subsection{%
2230     \@tempdima\jsc@tocl@width \advance\@tempdima -1zw
2231     \@dottedtocline{2}{\@tempdima}{3zw}}
2232 \newcommand*\l@subsubsection{%
2233     \@tempdima\jsc@tocl@width \advance\@tempdima 0zw
2234     \@dottedtocline{3}{\@tempdima}{4zw}}
2235 \newcommand*\l@paragraph{%
2236     \@tempdima\jsc@tocl@width \advance\@tempdima 1zw
2237     \@dottedtocline{4}{\@tempdima}{5zw}}
2238 \newcommand*\l@subparagraph{%
2239     \@tempdima\jsc@tocl@width \advance\@tempdima 2zw
2240     \@dottedtocline{5}{\@tempdima}{6zw}}
2241 \!/book&!report)
2242 (*book | report)
2243 % \newcommand*\l@subsection {\@dottedtocline{2}{3.8em}{3.2em}}
2244 % \newcommand*\l@subsubsection{\@dottedtocline{3}{7.0em}{4.1em}}
2245 % \newcommand*\l@paragraph {\@dottedtocline{4}{10em}{5em}}
2246 % \newcommand*\l@subparagraph {\@dottedtocline{5}{12em}{6em}}
2247 \newcommand*\l@section{%
2248     \@tempdima\jsc@tocl@width \advance\@tempdima -1zw
2249     \@dottedtocline{1}{\@tempdima}{3.683zw}}
2250 \newcommand*\l@subsection{%
2251     \@tempdima\jsc@tocl@width \advance\@tempdima 2.683zw
2252     \@dottedtocline{2}{\@tempdima}{3.5zw}}
2253 \newcommand*\l@subsubsection{%
2254     \@tempdima\jsc@tocl@width \advance\@tempdima 6.183zw
2255     \@dottedtocline{3}{\@tempdima}{4.5zw}}
2256 \newcommand*\l@paragraph{%
2257     \@tempdima\jsc@tocl@width \advance\@tempdima 10.683zw
2258     \@dottedtocline{4}{\@tempdima}{5.5zw}}
2259 \newcommand*\l@subparagraph{%
2260     \@tempdima\jsc@tocl@width \advance\@tempdima 16.183zw
2261     \@dottedtocline{5}{\@tempdima}{6.5zw}}
2262 \!/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
`\@lnumwidth` すが、アスキー版では `\@tempdima` の代わりに `\@lnumwidth` という変数で幅を決めるよう
に再定義しています。後続文字が全角か半角かでスペースが変わらないように `\hspace` を
入れておきました。

```

2263 \newdimen\@lnumwidth
2264 \def\numberline#1{\hb@xt@\@lnumwidth{#1\hfil}\hspace{0pt}}

```


`\@dottedtocline` L^AT_EX 本体 (`ltsect.dtx` 参照) での定義と同じですが, `\@tempdima` を `\@lnumwidth` に
`\jsTocLine` 変えています。

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

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

とします。

```
2265 \def\jsTocLine{\leaders\hbox{%
2266   $\m@th \mkern \@dotsep mu\hbox{.}\mkern \@dotsep mu$}\hfill}
2267 \def\@dottedtocline#1#2#3#4#5{\ifnum #1>\c@tocdepth \else
2268   \vskip \z@ \@plus.2\jsc@empt
2269   {\leftskip #2\relax \rightskip \@tocrmarg \parfillskip -\rightskip
2270    \parindent #2\relax\@afterindenttrue
2271    \interlinepenalty\@M
2272    \leavevmode
2273    \@lnumwidth #3\relax
2274    \advance\leftskip \@lnumwidth \null\nobreak\hskip -\leftskip
2275     {#4}\nobreak
2276     \jsTocLine \nobreak\hb@xt@\@pnumwidth{%
2277       \hfil\normalfont \normalcolor #5}\par}\fi}
```

■ 図目次と表目次

`\listoffigures` 図目次を出力します。

```
2278 \newcommand{\listoffigures}{%
2279 (*book | report)
2280   \if@twocolumn\@restonecoltrue\onecolumn
2281   \else\@restonecolfalse\fi
2282   \chapter*{\listfigurename}%
2283   \@mkboth{\listfigurename}{}%
2284   </book | report)
2285   (*!book&!report)
2286   \section*{\listfigurename}%
2287   \@mkboth{\listfigurename}{\listfigurename}%
2288   </!book&!report)
2289   \@starttoc{lof}%
2290   (book | report) \if@restonecol\twocolumn\fi
2291 }
```

`\l@figure` 図目次の項目を出力します。

```
2292 \newcommand*{\l@figure}{\@dottedtocline{1}{1zw}{3.683zw}}
```

`\listoftables` 表目次を出力します。

```
2293 \newcommand{\listoftables}{%
2294 (*book | report)
2295   \if@twocolumn\@restonecoltrue\onecolumn
```

```

2296 \else\@restonecolfalse\fi
2297 \chapter*{\listtablename}%
2298 \@mkboth{\listtablename}{}%
2299 </book | report>
2300 <!*book&!report>
2301 \section*{\listtablename}%
2302 \@mkboth{\listtablename}{\listtablename}%
2303 </!book&!report>
2304 \@starttoc{lot}%
2305 (book | report) \if@restonecol\twocolumn\fi
2306 }

```

`\l@table` 表目次は図目次と同じです。

```
2307 \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。

```

2308 \newdimen\bibindent
2309 \setlength\bibindent{2zw}

```

`thebibliography` 参考文献リストを出力します。

```

2310 \newenvironment{thebibliography}[1]{%
2311 \global\let\presectionname\relax
2312 \global\let\postsectionname\relax
2313 (article | jspf) \section*{\refname}\@mkboth{\refname}{\refname}%
2314 <*kiyou>
2315 \vspace{1.5\baselineskip}
2316 \subsubsection*{\refname}\@mkboth{\refname}{\refname}%
2317 \vspace{0.5\baselineskip}
2318 </kiyou>
2319 (book | report) \chapter*{\bibname}\@mkboth{\bibname}{}%
2320 (book | report) \addcontentsline{toc}{chapter}{\bibname}%
2321 \list{\@biblabel{\@arabic\c@enumiv}}%
2322 {\settowidth\labelwidth{\@biblabel{#1}}%
2323 \leftmargin\labelwidth
2324 \advance\leftmargin\labelsep
2325 \@openbib@code
2326 \usecounter{enumiv}%
2327 \let\p@enumiv\@empty
2328 \renewcommand\theenumiv{\@arabic\c@enumiv}}%
2329 (kiyou) \small
2330 \sloppy
2331 \clubpenalty4000
2332 \@clubpenalty\clubpenalty
2333 \widowpenalty4000%
2334 \sfcode'\.\@m}
2335 {\def\@noitemerr

```

```
2336     {\@latex@warning{Empty ‘thebibliography’ environment}}%
2337     \endlist}
```

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

```
2338 \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。

```
2339 \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。

```
2340 % \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
 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。

```
2341 % \def\@citex[#1]#2{\leavevmode
2342 %     \let\@citea\empty
2343 %     \@cite{\@for\@citeb:=#2\do
2344 %         {\@citea\def\@citea{, \inhibitglue\penalty\@m\ }%
2345 %         \edef\@citeb{\expandafter\@firstofone\@citeb\@empty}}%
2346 %         \if@filesw\immediate\write\@auxout{\string\citation{\@citeb}}\fi
2347 %         \@ifundefined{b@\@citeb}{\mbox{\normalfont\bfseries ?}}%
2348 %         \G@refundefinedtrue
2349 %         \@latex@warning
2350 %             {Citation ‘\@citeb’ on page \thepage \space undefined}}%
2351 %         {\@cite@ofmt{\csname b@\@citeb\endcsname}}}{#1}}
2352 % \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。

```
2353 % \DeclareRobustCommand\cite{\unskip
2354 %     \@ifnextchar [{\@tempwatruel\@citex}{\@tempwafalse\@citex[]}]
2355 %     \def\@cite#1#2{${\hbox{\scriptsize{#1\if@tempswa
2356 %         , \inhibitglue\ #2\fi} }}$}
```

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ときにマージンがずれる現象を直しました (Thanks: 藤村さん)。

```
2357 \newenvironment{theindex}{% 索引を 3 段組で出力する環境
2358     \if@twocolumn
2359         \onecolumn\@restonecolfalse
```

```

2360 \else
2361 \clearpage\@restonecoltrue
2362 \fi
2363 \columnseprule.4pt \columnsep 2zw
2364 \ifx\multicols\@undefined
2365 (book | report) \twocolumn[\@makeschapterhead{\indexname}]%
2366 (book | report) \addcontentsline{toc}{chapter}{\indexname}]%
2367 (!book&!report) \def\presectionname{}\def\postsectionname{}%
2368 (!book&!report) \twocolumn[\section*{\indexname}]%
2369 \else
2370 \ifdim\textwidth<\fullwidth
2371 \setlength{\evensidemargin}{\oddsidemargin}
2372 \setlength{\textwidth}{\fullwidth}
2373 \setlength{\linewidth}{\fullwidth}
2374 (book | report) \begin{multicols}{3}[\chapter*{\indexname}]%
2375 (book | report) \addcontentsline{toc}{chapter}{\indexname}]%
2376 (!book&!report) \def\presectionname{}\def\postsectionname{}%
2377 (!book&!report) \begin{multicols}{3}[\section*{\indexname}]%
2378 \else
2379 (book | report) \begin{multicols}{2}[\chapter*{\indexname}]%
2380 (book | report) \addcontentsline{toc}{chapter}{\indexname}]%
2381 (!book&!report) \def\presectionname{}\def\postsectionname{}%
2382 (!book&!report) \begin{multicols}{2}[\section*{\indexname}]%
2383 \fi
2384 \fi
2385 (book | report) \@mkboth{\indexname}{}%
2386 (!book&!report) \@mkboth{\indexname}{\indexname}%
2387 \plainifnotempty % \thispagestyle{plain}
2388 \parindent\z@
2389 \parskip\z@ \@plus .3\jcs@mp\relax
2390 \let\item\@idxitem
2391 \raggedright
2392 \footnotesize\narrowbaselines
2393 }{
2394 \ifx\multicols\@undefined
2395 \if@restonecol\onecolumn\fi
2396 \else
2397 \end{multicols}
2398 \fi
2399 \clearpage
2400 }

```

`\@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 2401 \newcommand{\@idxitem}{\par\hangindent 4zw} % 元 40pt
\subsubitem 2402 \newcommand{\subitem}{\@idxitem \hspace*{2zw}} % 元 20pt
2403 \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。

```

2404 \newcommand{\indexspace}{\par \vskip 10\jcs@mp \@plus5\jcs@mp \@minus3\jcs@mp\relax}

```

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

```
2405 \newcommand\seename{\if@english see\else →\fi}
2406 \newcommand\alsoname{\if@english see also\else →\fi}
```

10.4 脚注

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

```
2407 \@ifl@t@r\pfmtversion{2016/09/03}
2408   {\jsc@needspace@tchfalse}{\jsc@needspace@tchtrue}
2409 \ifjsc@needspace@tch
2410   \let\footnotes@ve=\footnote
2411   \def\footnote{\inhibitglue\footnotes@ve}
2412   \let\footnotemarks@ve=\footnotemark
2413   \def\footnotemark{\inhibitglue\footnotemarks@ve}
2414 \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。

```
2415 \@ifl@t@r\pfmtversion{2016/04/17}
2416   {\jsc@needspace@tchfalse}{\jsc@needspace@tchtrue}
2417 \ifjsc@needspace@tch
2418 \renewcommand\@makefnmark{%
2419   \ifysdir \hbox{ }\hbox{\@textsuperscript{\normalfont\@thefnmark}}\hbox{ }%
2420   \else \hbox{\yoko\@textsuperscript{\normalfont\@thefnmark}}\fi}
2421 \fi
```

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

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

[2016-10-08] TODO: 脚注番号が `newttext` や `newpertext` の使用時におかしくなっています。これらのパッケージは内部で `\thefootnote` を再定義していますので, 気になる

場合はパッケージを読み込むときに `defaultsup` オプションを付けてください (qa:57284, qa:57287)。

```
2422 \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。

```
2423 % \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。

```
2424 \renewcommand{\footnoterule}{%
2425   \kern-3\jsc@mp
2426   \hrule width .4\columnwidth height 0.4\jsc@mp
2427   \kern 2.6\jsc@mp}
```

`\c@footnote` 脚注番号は章ごとにリセットされます。

```
2428 (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_TE_X の「閉じ括弧類の直後に `\@footnotetext` が続く場合に改行が起きることがある問題に対処」と同等のコードを追加しました。

[2016-09-08] コミュニティ版 p_LA_TE_X のバグ修正に追随しました。

[2016-11-29] 古い p_LA_TE_X で使用された場合を考慮してコードを改良。

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

```
2429 \long\def\@footnotetext{%
2430   \insert\footins\group
2431   \normalfont\footnotesize
2432   \interlinepenalty\interfootnotelinepenalty
2433   \splittopskip\footnotesep
2434   \splitmaxdepth \dp\strutbox \floatingpenalty \@MM
2435   \hsize\columnwidth \@parboxrestore
2436   \protected@edef\@currentlabel{%
2437     \csname p@footnote\endcsname\@thefnmark
2438   }%
2439   \color@begingroup
2440   \@makefnmark{%
2441     \rule\z@\footnotesep\ignorespaces}%
2442   \futurelet\jsc@next\jsc@fo@t}
2443 \def\jsc@fo@t{\ifcat\group\noexpand\jsc@next \let\jsc@next\jsc@fo@t
2444               \else \let\jsc@next\jsc@fo@t\fi \jsc@next}
2445 \def\jsc@fo@t{\bgroup\aftergroup\jsc@@foot\let\jsc@next}
2446 \def\jsc@fo@t#1{\#1\jsc@@foot}
2447 \def\jsc@@foot{\@finalstrut\strutbox\color@endgroup\egroup
2448   \ifx\pltx@foot@penalty\undefined\else
2449     \ifhmode\null\fi
2450     \ifnum\pltx@foot@penalty=\z@\else
2451       \penalty\pltx@foot@penalty
```

```

2452     \pltx@foot@penalty\z@
2453     \fi
2454     \fi}

```

`\@makefntext` 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。

```

2455 \newcommand\@makefntext[1]{%
2456   \advance\leftskip 3zw
2457   \parindent 1zw
2458   \noindent
2459   \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。

```

2460 % \def\@xfootnotenext[#1]{%
2461 %   \begingroup
2462 %     \ifnum#1>\z@
2463 %       \csname c@\@mpfn\endcsname #1\relax
2464 %       \unrestored@protected@xdef\@thefnmark{\thempfn}%
2465 %     \else
2466 %       \unrestored@protected@xdef\@thefnmark{}%
2467 %     \fi
2468 %   \endgroup
2469 %   \@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。間違っても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)。

```

2470 \def\@inhibitglue{%
2471   \futurelet\@let@token\@@inhibitglue}
2472 \begingroup
2473 \let\GDEF=\gdef
2474 \let\CATCODE=\catcode
2475 \let\ENDGROUP=\endgroup

```

```

2476 \CATCODE'k=12
2477 \CATCODE'a=12
2478 \CATCODE'n=12
2479 \CATCODE'j=12
2480 \CATCODE'i=12
2481 \CATCODE'c=12
2482 \CATCODE'h=12
2483 \CATCODE'r=12
2484 \CATCODE't=12
2485 \CATCODE'e=12
2486 \GDEF\KANJI@CHARACTER{kanji character }
2487 \ENDGROUP
2488 \def\@@inhibitglue{%
2489   \expandafter\expandafter\expandafter\jsc@inhibitglue\expandafter\meaning\expandafter\@let@t
2490 \expandafter\def\expandafter\jsc@inhibitglue\expandafter#\expandafter1\KANJI@CHARACTER#2#3\jsc
2491   \def\jsc@ig@temp{#1}%
2492   \ifx\jsc@ig@temp@empty
2493     \ifnum\the\inhibitxspcode'#2=2\relax
2494       \inhibitglue
2495     \fi
2496   \fi}
2497 \let\everyparhook=\@inhibitglue
2498 \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。

```

2499 \def\@doendpe{%
2500   \@endpetrue
2501   \def\par{%
2502     \@restorepar\clubpenalty\@clubpenalty\everypar{\everyparhook}\par\@endpefalse}%
2503   \everypar{\setbox\z@\lastbox}\everypar{\everyparhook}\@endpefalse\everyparhook}}

```

[2017-08-31] `minipage` 環境にも対策します。

```

2504 \def\@setminipage{%
2505   \@minipagetrue
2506   \everypar{\@minipagefalse\everypar{\everyparhook}}%
2507 }

```

`\item` 命令の直後です。

```

2508 \def\@item[#1]{%
2509   \if@noperitem
2510     \@donoperitem
2511   \else
2512     \if@inlabel
2513       \indent \par
2514     \fi

```



```

2515 \ifhmode
2516 \unskip\unskip \par
2517 \fi
2518 \if@newlist
2519 \if@nobreak
2520 \@nbitem
2521 \else
2522 \addpenalty\@beginparpenalty
2523 \addvspace\@topsep
2524 \addvspace{-\parskip}%
2525 \fi
2526 \else
2527 \addpenalty\@itempenalty
2528 \addvspace\itemsep
2529 \fi
2530 \global\@inlabeltrue
2531 \fi
2532 \everypar{%
2533 \@minipagefalse
2534 \global\@newlistfalse
2535 \if@inlabel
2536 \global\@inlabelfalse
2537 {\setbox\z@\lastbox
2538 \ifvoid\z@
2539 \kern-\itemindent
2540 \fi}%
2541 \box\@labels
2542 \penalty\z@
2543 \fi
2544 \if@nobreak
2545 \@nobreakfalse
2546 \clubpenalty \@M
2547 \else
2548 \clubpenalty \@clubpenalty
2549 \everypar{\everyparhook}%
2550 \fi\everyparhook}%
2551 \if@noitemarg
2552 \@noitemargfalse
2553 \if@nmbrlist
2554 \refstepcounter\@listctr
2555 \fi
2556 \fi
2557 \sbox\@tempboxa{\makelabel{#1}}%
2558 \global\setbox\@labels\hbox{%
2559 \unhbox\@labels
2560 \hskip \itemindent
2561 \hskip -\labelwidth
2562 \hskip -\labelsep
2563 \ifdim \wd\@tempboxa >\labelwidth

```

```

2564     \box\@tempboxa
2565     \else
2566     \hbox to\labelwidth {\unhbox\@tempboxa}%
2567     \fi
2568     \hskip \labelsep}%
2569     \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。

```

2570 \def\@afterheading{%
2571   \@nobreaktrue
2572   \everypar{%
2573     \if@nobreak
2574       \@nobreakfalse
2575       \clubpenalty \@M
2576     \if@afterindent \else
2577       {\setbox\z@\lastbox}%
2578     \fi
2579   \else
2580     \clubpenalty \@clubpenalty
2581     \everypar{\everyparhook}%
2582   \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。

```

2583 \def\@gnewline #1{%
2584   \ifvmode
2585     \@nolnerr
2586   \else
2587     \unskip \reserved@e {\reserved@f#1}\nobreak \hfil \break \null
2588     \inhibitglue \ignorespaces
2589   \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。

```

2590 \if@jslogo
2591   \IfFileExists{jslogo.sty}{%

```

```

2592 \RequirePackage{jslogo}%
2593 \def\小{\jslg@small}%
2594 \def\上小{\jslg@uppersmall}%
2595 }{%
2596 \ClassWarningNoLine{\jsc@clsname}{%
2597   The redefinitions of LaTeX-related logos has\MessageBreak
2598   been moved to jslogo.sty since 2016, but\MessageBreak
2599   jslogo.sty not found. Current release of\MessageBreak
2600   'jsclasses' includes it, so please check\MessageBreak
2601   the installation}%
2602 }
2603 \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` 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。

```

2604 %\let\ltx@ifnextchar\@ifnextchar
2605 %\def\ProvidesFile#1{%
2606 % \begingroup
2607 %   \catcode'\ 10 %
2608 %   \ifnum \endlinechar<256 %
2609 %     \ifnum \endlinechar>\m@ne
2610 %       \catcode\endlinechar 10 %
2611 %     \fi
2612 %   \fi
2613 %   \@makeoother\/%
2614 %   \@makeoother\&%
2615 %   \ltx@ifnextchar[{\@providesfile{#1}}{\@providesfile{#1}[]}]

```

14 初期設定

■いろいろな語

```

\prepartname
\postpartname 2616 \newcommand{\prepartname}{\if@english Part~\else 第 \fi}
\prechaptername 2617 \newcommand{\postpartname}{\if@english \else 部 \fi}
\postchaptername 2618 (book | report)\newcommand{\prechaptername}{\if@english Chapter~\else 第 \fi}
\presectionname 2619 (book | report)\newcommand{\postchaptername}{\if@english \else 章 \fi}
\postsectionname

```

```

2620 \newcommand{\presectionname}{}% 第
2621 \newcommand{\postsectionname}{}% 節

\contentsname
\listfigurename 2622 \newcommand{\contentsname}{\if@english Contents\else 目次 \fi}
\listtablename 2623 \newcommand{\listfigurename}{\if@english List of Figures\else 図目次 \fi}
                2624 \newcommand{\listtablename}{\if@english List of Tables\else 表目次 \fi}

\refname
\bibname 2625 \newcommand{\refname}{\if@english References\else 参考文献 \fi}
\indexname 2626 \newcommand{\bibname}{\if@english Bibliography\else 参考文献 \fi}
           2627 \newcommand{\indexname}{\if@english Index\else 索引 \fi}

\figurename
\tablename 2628 \!jspf\newcommand{\figurename}{\if@english Fig.~\else 図 \fi}
           2629 \jspf\newcommand{\figurename}{Fig.~}
           2630 \!jspf\newcommand{\tablename}{\if@english Table~\else 表 \fi}
           2631 \jspf\newcommand{\tablename}{Table~}

\appendixname
\abstractname 2632 % \newcommand{\appendixname}{\if@english Appendix~\else 付録 \fi}
           2633 \newcommand{\appendixname}{\if@english \else 付録 \fi}
           2634 \!book\newcommand{\abstractname}{\if@english Abstract\else 概要 \fi}

    ■今日の日付 LATEX で処理した日付を出力します。和暦にするには \和暦 と書いてくださ
    い。ちなみにこの文章の作成日は西暦では 2020 年 2 月 3 日で、和暦では令和 2 年 2 月 3 日
    です。

\today
2635 \newif\if 西暦 \西暦 true
2636 \def\西暦{\西暦 true}
2637 \def\和暦{\西暦 false}
2638 \newcount\heisei \heisei\year \advance\heisei-1988\relax
2639 \def\pltx@today@year@#1{%
2640   \ifnum\numexpr\year-#1=1 元 \else
2641     \ifnum1=\iftdir\ifmdir0\else1\fi\else0\fi
2642     \kansuji\numexpr\year-#1\relax
2643   \else
2644     \number\numexpr\year-#1\relax\nobreak
2645   \fi
2646 \fi 年
2647 }
2648 \def\pltx@today@year{%
2649   \ifnum\numexpr\year*10000+\month*100+\day<19890108
2650     昭和 \pltx@today@year@{1925}%
2651   \else\ifnum\numexpr\year*10000+\month*100+\day<20190501
2652     平成 \pltx@today@year@{1988}%
2653   \else

```

```

2654  令和 \pltx@today@year@{2018}%
2655  \fi\fi}
2656  \def\today{%
2657  \if@english
2658    \ifcase\month\or
2659    January\or February\or March\or April\or May\or June\or
2660    July\or August\or September\or October\or November\or December\fi
2661    \space\number\day, \number\year
2662  \else\if 西暦
2663    \ifnum1=\iftdir\ifmdir0\else1\fi\else0\fi \kansuji\year
2664    \else\number\year\nobreak\fi 年
2665  \else
2666    \pltx@today@year
2667  \fi
2668  \ifnum1=\iftdir\ifmdir0\else1\fi\else0\fi
2669    \kansuji\month 月
2670    \kansuji\day 日
2671  \else
2672    \number\month\nobreak 月
2673    \number\day\nobreak 日
2674  \fi\fi}

```

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

```

2675  \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。

```

2676  (article)\if@slide \pagestyle{empty} \else \pagestyle{plain} \fi
2677  (book)\if@report \pagestyle{plain} \else \pagestyle{headings} \fi
2678  (report|kiyou)\pagestyle{plain}
2679  (jspf)\pagestyle{headings}
2680  \pagenumbering{arabic}
2681  \if@twocolumn
2682    \twocolumn
2683    \sloppy
2684    \flushbottom
2685  \else
2686    \onecolumn
2687    \raggedbottom
2688  \fi
2689  \if@slide
2690    \renewcommand\kanjifamilydefault{\gtdefault}
2691    \renewcommand?familydefault{\sfdefault}
2692    \raggedright
2693    \xkanjiskip=0.1em\relax
2694  \fi

```

15 実験的コード

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

```
2695 (*article | book | report)
2696 \ifpackageloaded{exppl2e}{\jsc@needspace@tchtrue}{\jsc@needspace@tchfalse}
2697 \ifjsc@needspace@tch\else
2698   \expandafter\endinput
2699 \fi
```

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

```
\@gnewline
2700 \def\@gnewline #1{%
2701   \ifvmode
2702     \@nolnerr
2703   \else
2704     \unskip \reserved@a {\reserved@f#1}\nobreak \hfil \break \hskip \z@
2705     \ignorespaces
2706   \fi}
2707 </article | book | report)
```

以上です。