

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

2025/04/27

1 はじめに

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

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

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

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

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

ここでは次のドキュメントクラス (スタイルファイル) を作ります。

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

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

L^AT_EX 2_ε あるいは pL^AT_EX 2_ε 標準のドキュメントクラスとの違いを説明しておきます。

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

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

のように `mingoth` オプションを付けます。

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

以下では実際のコードに即して説明します。

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

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

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

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

`\ifjsc@needspatch` [2016-08-22] 従来 `jsclasses` では、 $\text{p}\text{\LaTeX}$ や $\text{L}\text{\LaTeX}$ 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 $\text{p}\text{\LaTeX}$ 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 $\text{p}\text{\LaTeX}$ カーネルと衝突しないように、日付が古い場合だけパッチをあてる場合があります。この処理に使用するフラグを定義します。

```
16 </class>
17 <(*class | minijs)
18 \newif\ifjsc@needspatch
19 \jsc@needspatchfalse
20 </class | minijs>
21 <(*class>
```

2 オプション

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

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

`\if@restonecol` 段組のときに真になる論理変数です。

```
22 \newif\if@restonecol
```

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

```
23 \newif\if@titlepage
```

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

```
24 <book | report>\newif\if@openright
```

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

```
25 <book | report>\newif\if@openleft
```

`\if@mainmatter` 真なら本文, 偽なら前付け・後付けです。偽なら `\chapter` で章番号が出ません。

```
26 <book>\newif\if@mainmatter \@mainmattertrue
```

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

```
27 \newif\if@enablejfam \@enablejfamtrue
```

以下で各オプションを宣言します。

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

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

デフォルトは `a4paper` です。

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

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

```

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

```

■横置き 用紙の縦と横の長さを入れ換えます。

```

76 \newif\if@landscape
77 \@landscapefalse
78 \DeclareOption{landscape}{\@landscapetrue}

```

■slide オプション slide を新設しました。

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

```

79 \newif\if@slide
80 \@slidefalse

```

■サイズオプション 10pt, 11pt, 12pt のほかに, 8pt, 9pt, 14pt, 17pt, 21pt, 25pt, 30pt, 36pt, 43pt を追加しました。これは等比数列になるように選んだものです（従来の

20pt も残しました)。`\@ptsize` の定義が変だったのでご迷惑をおかけしましたが、標準的なドキュメントクラスと同様にポイント数から 10 を引いたものに直しました。

[2003-03-22] 14Q オプションを追加しました。

[2003-04-18] 12Q オプションを追加しました。

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

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

```

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

```

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

```

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

```

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

```

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

```

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

```

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

```

■二段組 `twocolumn` で二段組になります。

```

134 \DeclareOption{onecolumn}{\@twocolumnfalse}
135 \DeclareOption{twocolumn}{\@twocolumntrue}

```

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

```

136 \DeclareOption{titlepage}{\@titlepagetrue}
137 \DeclareOption{notitlepage}{\@titlepagefalse}

```

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

[2017-02-24] `openright` は横組では奇数ページ起こし，縦組では偶数ページ起こしを表します。ややこしいですが，これは \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。

```

138 <book | report>\DeclareOption{openright}{\@openrighttrue\@openleftfalse}

```

```

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

```

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

eqnarray (env.) L^AT_EX 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。

[2022-09-13] L^AT_EX 2_ε 2021-11-15 (ltmath.dtx 2021/10/14 v1.2j) で \@currentcounter が追加されましたので、追従します。

```

141 \def\eqnarray{%
142   \stepcounter{equation}%
143   \def\@currentlabel{\p@equation\theequation}%
144   \def\@currentcounter{equation}%
145   \global\@eqnswtrue
146   \m@th
147   \global\@eqcnt\z@
148   \tabskip\@centering
149   \let\@eqncr
150   $$\everycr{}\halign to\displaywidth\bgroup
151     \hskip\@centering$\displaystyle\tabskip\z@skip{##}$\@eqnse1
152     &\global\@eqcnt\@ne \hfil$\displaystyle{##}$\hfil
153     &\global\@eqcnt\tw@ $\displaystyle{##}$\hfil\tabskip\@centering
154     &\global\@eqcnt\thr@@ \hb@xt@\z@\bgroup\hss##\egroup
155     \tabskip\z@skip
156   \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。

[2022-09-13] L^AT_EX 2_ε 2021-11-15 (ltmath.dtx 2021/10/14 v1.2j) で \@currentcounter が追加されましたので、追従します。

```

157 \DeclareOption{leqno}{\input{leqno.clo}}
158 \DeclareOption{fleqn}{\input{fleqn.clo}}
159 % fleqn 用の eqnarray 環境の再定義
160 \def\eqnarray{%
161   \stepcounter{equation}%
162   \def\@currentlabel{\p@equation\theequation}%
163   \def\@currentcounter{equation}%
164   \global\@eqnswtrue\m@th
165   \global\@eqcnt\z@
166   \tabskip\mathindent
167   \let\@eqncr
168   \setlength\abovedisplayskip{\topsep}%
169   \ifvmode
170     \addtolength\abovedisplayskip{\partopsep}%
171   \fi
172   \addtolength\abovedisplayskip{\parskip}%
173   \setlength\belowdisplayskip{\abovedisplayskip}%
174   \setlength\belowdisplayshortskip{\abovedisplayskip}%

```

```

175 \setlength\abovedisplaysshortskip{\abovedisplayskip}%
176 $$\everycr{}\halign to\linewidth% $$
177 \bgroup
178 \hskip\@centering$\displaystyle\tabskip\z@skip{##}$\@eqnse
179 &\global\@eqcnt\@ne \hfil$\displaystyle{#{##}}$\hfil
180 &\global\@eqcnt\tw@
181 $\displaystyle{##}$\hfil \tabskip\@centering
182 &\global\@eqcnt\thr@@ \hb@xt@\z@\bgroup\hss##\egroup
183 \tabskip\z@skip\cr
184 }}

```

■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。

```

185 % \DeclareOption{openbib}{%
186 % \AtEndOfPackage{%
187 % \renewcommand\@openbib@code{%
188 % \advance\leftmargin\bibindent
189 % \itemindent -\bibindent
190 % \listparindent \itemindent
191 % \parsep \z@}%
192 % \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。

```

193 \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。

```

194 \DeclareOption{draft}{\setlength\overfullrule{5pt}}
195 \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, dvi2pdfmx 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。

```
196 \newif\ifmingoth
197 \mingothfalse
198 \newif\ifjisfont
199 \jisfontfalse
200 \newif\if@jsc@uplatex
201 \@jsc@uplatexfalse
202 \DeclareOption{winjis}{%
203     \ClassWarningNoLine{\jsc@clsname}{%
204         The option 'winjis' has been removed;\MessageBreak
205         Use '\string\usepackage{winjis}' instead}}
206 \DeclareOption{mingoth}{\mingothtrue}
207 \DeclareOption{jis}{\jisfonttrue}
208 \let\jsc@engine@given=*
209 \DeclareOption{platex}{\let\jsc@engine@given=p\@jsc@uplatexfalse}
210 \DeclareOption{uplatex}{\let\jsc@engine@given=u\@jsc@uplatextrue}
211 \DeclareOption{autodetect-engine}{\let\jsc@engine@given=*}
212 \def\jsc@JYn{\if@jsc@uplatex JY2\else JY1\fi}
213 \def\jsc@JTn{\if@jsc@uplatex JT2\else JT1\fi}
214 \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。

```
215 \newif\ifpapersize
216 \papersizefalse
217 \DeclareOption{papersize}{\papersizetrue}
```

■英語化 オプション english を新設しました。

```
218 \newif\if@english
219 \@englishfalse
220 \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。

```
221 <*book>
222 \newif\if@report
223 \@reportfalse
224 \DeclareOption{report}{\@reporttrue\@openrightfalse\@twosidefalse\@mparswitchfalse}
225 </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。

```
226 \newif\if@jslogo \@jslogotrue
227 \DeclareOption{jslogo}{\@jslogotrue}
228 \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。

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

後処理

```
235 \if@slide
236   \def\maybeblue{\@ifundefined{ver@color.sty}{\color{blue}}{}}
237 \fi
238 \if@landscape
239   \setlength\@tempdima {\paperheight}
240   \setlength\paperheight{\paperwidth}
241   \setlength\paperwidth {\@tempdima}
242 \fi
```

■使用エンジンの検査・自動判定 デフォルトで現在使われているエンジンが pL^AT_EX か upL^AT_EX かを判定します。ユーザによって platex オプションまたは uplatex オプションが明示的に指定されている場合は、実際に使われているエンジンと一致しているかを検査し、一致しない場合はエラーメッセージを表示します。

[2016-11-09] pL^AT_EX/ upL^AT_EX を自動判別するオプション autodetect-engine を新設しました。upL^AT_EX 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。

[2023-02-12] autodetect-engine 指定時の挙動を規定化しました。また platex を新設しました。オプション autodetect-engine, platex, uplatex のうち最後に指定されたものが有効になります。

```
243 \ifnum \ifx\ucs\@undefined\z@\else\ucs"3000 \fi ="3000
244   \ifx *\jsc@engine@given
245     \ClassInfo\jsc@clsname{Autodetected engine: upLaTeX}
246     \@jsc@uplatextrue
247     \g@addto@macro\@classoptionslist{,uplatex}
248   \fi
249   \ifx p\jsc@engine@given
250     \ClassError\jsc@clsname
251       {Option 'platex' is specified but you are running upLaTeX.}
252     {\@ehc}
253   \fi
254   \if@jsc@uplatex\else
255     \ClassError\jsc@clsname
256       {You are running upLaTeX.\MessageBreak
257         Please use pLaTeX instead, or remove 'platex' from\MessageBreak
258         the class option list}
259     {\@ehc}
```

```

260 \jsc@uplatextrue
261 \fi
262 \else
263 \ifx *\jsc@engine@given
264 \ClassInfo\jsc@clsname{Autodetected engine: pLaTeX}
265 \jsc@uplatexfalse
266 \g@addto@macro\@classoptionslist{,pllatex}
267 \fi
268 \ifx u\jsc@engine@given
269 \ClassError\jsc@clsname
270 {Option 'uplatex' is specified but you are running pLaTeX.}
271 {\@ehc}
272 \fi
273 \if@jsc@uplatex
274 \ClassError\jsc@clsname
275 {You are running pLaTeX.\MessageBreak
276 Please use upLaTeX instead, or remove 'uplatex' from\MessageBreak
277 the class option list}
278 {\@ehc}
279 \jsc@uplatexfalse
280 \fi
281 \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 dvipout 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 pLaTeX 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。

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

[2022-09-12] 次期 LaTeX 2_ε カーネルに \stockwidth, \stockheight が追加されるよう

です。クラスファイル側では未定義のときのみこれらの長さ変数を定義します。h20y6m
さん、ありがとうございます。

```
282 \iftombow
283   \ifx\stockwidth\@undefined\newdimen\stockwidth\fi
284   \ifx\stockheight\@undefined\newdimen\stockheight\fi
285   \setlength{\stockwidth}{\paperwidth}
286   \setlength{\stockheight}{\paperheight}
287   \advance \stockwidth 2in
288   \advance \stockheight 2in
289 \fi
290 \ifpapersize
291   \iftombow
292     \edef\jsc@papersize@special{papersize=\the\stockwidth,\the\stockheight}
293   \else
294     \edef\jsc@papersize@special{papersize=\the\paperwidth,\the\paperheight}
295   \fi
296   \AtBeginDvi{\special{\jsc@papersize@special}}
297 \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。

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

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

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

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

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

```
301 \newdimen\jsc@empt
302 \newdimen\jsc@mmm
303 \def\inv@mag{1}
304 \ifjsc@mag
305   \jsc@empt=1\p@
306   \jsc@mmm=1mm
307   \ifnum\@ptsize=-2
308     \mag 833
309     \def\inv@mag{1.20048}
310     \def\n@baseline{15}%
```

```

311 \fi
312 \ifnum\@ptsize=-1
313 \mag 913 % formerly 900
314 \def\inv@mag{1.09529}
315 \def\n@baseline{15}%
316 \fi
317 \ifnum\@ptsize=1
318 \mag 1095 % formerly 1100
319 \def\inv@mag{0.913242}
320 \fi
321 \ifnum\@ptsize=2
322 \mag 1200
323 \def\inv@mag{0.833333}
324 \fi
325 \ifnum\@ptsize=4
326 \mag 1440
327 \def\inv@mag{0.694444}
328 \fi
329 \ifnum\@ptsize=7
330 \mag 1728
331 \def\inv@mag{0.578704}
332 \fi
333 \ifnum\@ptsize=10
334 \mag 2000
335 \def\inv@mag{0.5}
336 \fi
337 \ifnum\@ptsize=11
338 \mag 2074
339 \def\inv@mag{0.48216}
340 \fi
341 \ifnum\@ptsize=15
342 \mag 2488
343 \def\inv@mag{0.401929}
344 \fi
345 \ifnum\@ptsize=20
346 \mag 2986
347 \def\inv@mag{0.334896}
348 \fi
349 \ifnum\@ptsize=26
350 \mag 3583
351 \def\inv@mag{0.279096}
352 \fi
353 \ifnum\@ptsize=33
354 \mag 4300
355 \def\inv@mag{0.232558}
356 \fi
357 \ifnum\@ptsize=1200
358 \mag 923
359 \def\inv@mag{1.0834236}

```

```

360 \fi
361 \ifnum\@ptsize=1400
362 \mag 1077
363 \def\inv@mag{0.928505}
364 \fi
365 \ifnum\@ptsize=1001
366 \mag 1085
367 \def\inv@mag{0.921659}
368 \fi
369 \ifnum\@ptsize=1051
370 \mag 1139
371 \def\inv@mag{0.877963}
372 \fi
373 \ifnum\@ptsize=1101
374 \mag 1194
375 \def\inv@mag{0.837521}
376 \fi
377 \ifnum\@ptsize=1201
378 \mag 1302
379 \def\inv@mag{0.768049}
380 \fi
381 \else
382 \jsc@mpt=\jsc@magscale\p@
383 \jsc@mmm=\jsc@magscale mm
384 \def\inv@mag{1}
385 \ifnum\@ptsize=-2
386 \def\n@baseline{15}%
387 \fi
388 \ifnum\@ptsize=-1
389 \def\n@baseline{15}%
390 \fi
391 \fi
392 <*kiyou>
393 \def\jsc@magscale{0.9769230}
394 \ifjsc@mag
395 \mag 977
396 \def\inv@mag{1.02354}
397 \jsc@mpt=1\p@
398 \jsc@mmm=1mm
399 \else
400 \jsc@mpt=\jsc@magscale\p@
401 \jsc@mmm=\jsc@magscale mm
402 \def\inv@mag{1}
403 \fi
404 </kiyou>
405 \ifjsc@mag@xreal
406 \RequirePackage{type1cm}
407 \mathchardef\jsc@csta=259
408 \def\jsc@invscale#1#2{%

```

```

409 \beginpgroup \@tempdima=#1\relax \@tempdimb#2\p@\relax
410 \@tempcnta\@tempdima \multiply\@tempcnta\@cclvi
411 \divide\@tempcnta\@tempdimb \multiply\@tempcnta\@cclvi
412 \@tempcntb\p@ \divide\@tempcntb\@tempdimb
413 \advance\@tempcnta-\@tempcntb \advance\@tempcnta-\tw@
414 \@tempdimb\@tempcnta\@ne
415 \advance\@tempcnta\@tempcntb \advance\@tempcnta\@tempcntb
416 \advance\@tempcnta\jsc@csta \@tempdimc\@tempcnta\@ne
417 \@whiledim\@tempdimb<\@tempdimc\do{%
418 \@tempcntb\@tempdimb \advance\@tempcntb\@tempdimc
419 \advance\@tempcntb\@ne \divide\@tempcntb\tw@
420 \ifdim #2\@tempcntb>\@tempdima
421 \advance\@tempcntb\m@ne \@tempdimc=\@tempcntb\@ne
422 \else \@tempdimb=\@tempcntb\@ne \fi}%
423 \xdef\jsc@gtmpa{\the\@tempdimb}%
424 \endpgroup #1=\jsc@gtmpa\relax}
425 \expandafter\let\csname OT1/cmr/m/n/10\endcsname\relax
426 \expandafter\let\csname OMX/cmex/m/n/10\endcsname\relax
427 \let\jsc@get@external@font\get@external@font
428 \def\get@external@font{%
429 \jsc@preadjust@extract@font
430 \jsc@get@external@font}
431 \def\jsc@fstrunc#1{%
432 \edef\jsc@tmpa{\strip@pt#1}%
433 \expandafter\jsc@fstrunc@a\jsc@tmpa.****\@nil}
434 \def\jsc@fstrunc@a#1.#2#3#4#5#6\@nil{%
435 \if#5* \else
436 \edef\jsc@tmpa{#1%
437 \ifnum#2#3>\z@ .#2\ifnum#3>\z@ #3\fi\fi}%
438 \fi}
439 \def\jsc@preadjust@extract@font{%
440 \let\jsc@req@size\f@size
441 \dimen@f@size\p@ \jsc@invscale\dimen@\jsc@magscale
442 \advance\dimen@.005pt\relax \jsc@fstrunc\dimen@
443 \let\jsc@ref@size\jsc@tmpa
444 \let\f@size\jsc@ref@size}
445 \def\execute@size@function#1{%
446 \let\jsc@cref@size\f@size
447 \let\f@size\jsc@req@size
448 \csname s@fct@#1\endcsname}
449 \let\jsc@DeclareErrorFont\DeclareErrorFont
450 \def\DeclareErrorFont#1#2#3#4#5{%
451 \@tempdimc#5\p@ \@tempdimc\jsc@magscale\@tempdimc
452 \edef\jsc@tmpa{#{1}{#2}{#3}{#4}{\strip@pt\@tempdimc}}
453 \expandafter\jsc@DeclareErrorFont\jsc@tmpa}
454 \def\gen@sfcnt{%
455 \edef\mandatory@arg{\mandatory@arg\jsc@cref@size}%
456 \empty@sfcnt}
457 \def\genb@sfcnt{%

```

```

458 \edef\mandatory@arg{%
459 \mandatory@arg\expandafter\genb@x\jsc@ceref@size..\@@}%
460 \empty@sfcnt}
461 \DeclareErrorFont{OT1}{cmr}{m}{n}{10}
462 \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 463 \def\jsc@smallskip{\vspace\jsc@smallskipamount}
\jsc@bigskip 464 %\def\jsc@medskip{\vspace\jsc@medskipamount}
465 %\def\jsc@bigskip{\vspace\jsc@bigskipamount}

```

`\jsc@smallskipamount`

```

\jsc@medskipamount 466 \newskip\jsc@smallskipamount
\jsc@bigskipamount 467 \jsc@smallskipamount=3\jsc@mpt plus 1\jsc@mpt minus 1\jsc@mpt
468 %\newskip\jsc@medskipamount
469 %\jsc@medskipamount =6\jsc@mpt plus 2\jsc@mpt minus 2\jsc@mpt
470 %\newskip\jsc@bigskipamount
471 %\jsc@bigskipamoun =12\jsc@mpt plus 4\jsc@mpt minus 4\jsc@mpt

```

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

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

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

```

472 \setlength\paperwidth{\inv@mag\paperwidth}%
473 \setlength\paperheight{\inv@mag\paperheight}%
474 \iftombow
475 \setlength\stockwidth{\inv@mag\stockwidth}%
476 \setlength\stockheight{\inv@mag\stockheight}%
477 \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。

```

478 % \ifpapersize
479 % \setlength{\@tempdima}{\paperwidth}
480 % \setlength{\@tempdimb}{\paperheight}
481 % \iftombow
482 % \advance \@tempdima 2truein
483 % \advance \@tempdimb 2truein
484 % \fi
485 % \AtBeginDvi{\special{pdf: pagesize width \the\@tempdima space height \the\@tempdimb}}
486 % \fi

```


3 和文フォントの変更

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

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

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

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

```
487 </class>
488 <*minijs>
489 %% min/goth -> jis/jisg (for pLaTeX only)
490 \ifnum\jis"2121="3000 \else
491 \for\@tempa:=5,6,7,8,9,10,10.95,12,14.4,17.28,20.74,24.88\do{%
492 \expandafter\let\csname JY1/mc/m/n/\@tempa\endcsname\relax
493 \expandafter\let\csname JY1/gt/m/n/\@tempa\endcsname\relax
494 \expandafter\let\csname JT1/mc/m/n/\@tempa\endcsname\relax
495 \expandafter\let\csname JT1/gt/m/n/\@tempa\endcsname\relax
496 }
497 \def\Cjascale{0.924690}
498 \DeclareFontShape{JY1}{mc}{m}{n}{<-> s * [0.961] jis}{}
499 \DeclareFontShape{JY1}{gt}{m}{n}{<-> s * [0.961] jisg}{}
500 \DeclareFontShape{JT1}{mc}{m}{n}{<-> s * [0.961] tmin10}{}
501 \DeclareFontShape{JT1}{gt}{m}{n}{<-> s * [0.961] tgoth10}{}
502 \fi
503 </minijs>
504 <*class>
505 <*!jspf>
506 \def\Cjascale{0.924690}
507 \ifmingoth
508 \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.961] \jsc@pfx@ min10}{}
509 \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.961] \jsc@pfx@ goth10}{}

```

```

510 \DeclareFontShape{\jsc@JTn}{mc}{m}{n}{<-> s * [0.961] \jsc@pfx@ tmin10}{ }
511 \DeclareFontShape{\jsc@JTn}{gt}{m}{n}{<-> s * [0.961] \jsc@pfx@ tgoth10}{ }
512 \else
513 \ifjisfont
514 \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.961] \jsc@pfx@ jis}{ }
515 \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.961] \jsc@pfx@ jisg}{ }
516 \DeclareFontShape{\jsc@JTn}{mc}{m}{n}{<-> s * [0.961] \jsc@pfx@ tmin10}{ }
517 \DeclareFontShape{\jsc@JTn}{gt}{m}{n}{<-> s * [0.961] \jsc@pfx@ tgoth10}{ }
518 \else
519 \if@jsc@uplatex
520 \DeclareFontShape{JY2}{mc}{m}{n}{<-> s * [0.924690] upjisr-h}{ }
521 \DeclareFontShape{JY2}{gt}{m}{n}{<-> s * [0.924690] upjisg-h}{ }
522 \DeclareFontShape{JT2}{mc}{m}{n}{<-> s * [0.924690] upjisr-v}{ }
523 \DeclareFontShape{JT2}{gt}{m}{n}{<-> s * [0.924690] upjisg-v}{ }
524 \else
525 \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.961] \jsc@pfx@ jis}{ }
526 \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.961] \jsc@pfx@ jisg}{ }
527 \DeclareFontShape{\jsc@JTn}{mc}{m}{n}{<-> s * [0.961] \jsc@pfx@ tmin10}{ }
528 \DeclareFontShape{\jsc@JTn}{gt}{m}{n}{<-> s * [0.961] \jsc@pfx@ tgoth10}{ }
529 \fi
530 \fi
531 \fi
532 \end{jspf}

```

某学会誌では、和文フォントを PostScript の 9 ポイントにするために、 $9/(9.62216 * 72/72.27) = 0.93885$ 倍します。

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

```

533 \begin{jspf}
534 \def\Cjascale{0.903375}
535 \ifmingoth
536 \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ min10}{ }
537 \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ goth10}{ }
538 \DeclareFontShape{\jsc@JTn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tmin10}{ }
539 \DeclareFontShape{\jsc@JTn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tgoth10}{ }
540 \else
541 \ifjisfont
542 \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ jis}{ }
543 \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ jisg}{ }
544 \DeclareFontShape{\jsc@JTn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tmin10}{ }
545 \DeclareFontShape{\jsc@JTn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tgoth10}{ }
546 \else
547 \if@jsc@uplatex
548 \DeclareFontShape{JY2}{mc}{m}{n}{<-> s * [0.903375] upjisr-h}{ }
549 \DeclareFontShape{JY2}{gt}{m}{n}{<-> s * [0.903375] upjisg-h}{ }
550 \DeclareFontShape{JT2}{mc}{m}{n}{<-> s * [0.903375] upjisr-v}{ }
551 \DeclareFontShape{JT2}{gt}{m}{n}{<-> s * [0.903375] upjisg-v}{ }
552 \else
553 \DeclareFontShape{\jsc@JYn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ jis}{ }
554 \DeclareFontShape{\jsc@JYn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ jisg}{ }

```

```

555 \DeclareFontShape{\jsc@JTn}{mc}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tmin10}{ }
556 \DeclareFontShape{\jsc@JTn}{gt}{m}{n}{<-> s * [0.93885] \jsc@pfx@ tgoth10}{ }
557 \fi
558 \fi
559 \fi
560 </jpsf>

```

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

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

[2004-11-03] `\rmfamily` も和文対応にしました。

```

561 % \DeclareFontShape{\jsc@JYn}{mc}{bx}{n}{<->ssub*gt/m/n}{ } % in \jsc@JYnmc
562 % \DeclareFontShape{\jsc@JYn}{gt}{bx}{n}{<->ssub*gt/m/n}{ } % in \jsc@JYngt
563 \DeclareFontShape{\jsc@JYn}{mc}{m}{it}{<->ssub*mc/m/n}{ }
564 \DeclareFontShape{\jsc@JYn}{mc}{m}{sl}{<->ssub*mc/m/n}{ }
565 \DeclareFontShape{\jsc@JYn}{mc}{m}{sc}{<->ssub*mc/m/n}{ }
566 \DeclareFontShape{\jsc@JYn}{gt}{m}{it}{<->ssub*gt/m/n}{ }
567 \DeclareFontShape{\jsc@JYn}{gt}{m}{sl}{<->ssub*gt/m/n}{ }
568 \DeclareFontShape{\jsc@JYn}{mc}{bx}{it}{<->ssub*gt/m/n}{ }
569 \DeclareFontShape{\jsc@JYn}{mc}{bx}{sl}{<->ssub*gt/m/n}{ }
570 % \DeclareFontShape{\jsc@JTn}{mc}{bx}{n}{<->ssub*gt/m/n}{ } % in \jsc@JTnmc
571 % \DeclareFontShape{\jsc@JTn}{gt}{bx}{n}{<->ssub*gt/m/n}{ } % in \jsc@JTngt
572 \DeclareFontShape{\jsc@JTn}{mc}{m}{it}{<->ssub*mc/m/n}{ }
573 \DeclareFontShape{\jsc@JTn}{mc}{m}{sl}{<->ssub*mc/m/n}{ }
574 \DeclareFontShape{\jsc@JTn}{mc}{m}{sc}{<->ssub*mc/m/n}{ }
575 \DeclareFontShape{\jsc@JTn}{gt}{m}{it}{<->ssub*gt/m/n}{ }
576 \DeclareFontShape{\jsc@JTn}{gt}{m}{sl}{<->ssub*gt/m/n}{ }
577 \DeclareFontShape{\jsc@JTn}{mc}{bx}{it}{<->ssub*gt/m/n}{ }
578 \DeclareFontShape{\jsc@JTn}{mc}{bx}{sl}{<->ssub*gt/m/n}{ }

```

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

[2020-10-04] $\mathrm{I}\mathrm{A}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ 2_ε 2020-10-01 では `\AddToHook` を利用します。

```

579 </class>
580 <*class | minijs>
581 %% ad-hoc "relation font"
582 \@ifl@t@r\fmtversion{2020/10/01}{
583   {\jsc@needsp@tchfalse}{\jsc@needsp@tchtrue}

```

```

584 \ifjsc@needspace % --- for 2020-02-02 or older BEGIN
585 \ifx\@rmfamilyhook\undefined % old
586 \DeclareRobustCommand\rmfamily
587     {\not@math@alphabet\rmfamily\mathrm
588         \romanfamily\rmdefault\kanjifamily\mcdefault\selectfont}
589 \DeclareRobustCommand\sffamily
590     {\not@math@alphabet\sffamily\mathsf
591         \romanfamily\sfdefault\kanjifamily\gtdefault\selectfont}
592 \DeclareRobustCommand\ttfamily
593     {\not@math@alphabet\ttfamily\mathtt
594         \romanfamily\ttdefault\kanjifamily\gtdefault\selectfont}
595 \AtBeginDocument{%
596     \ifx\mweights@init\undefined\else % mweights.sty is loaded
597         % my definitions above should have been overwritten, recover it!
598         % \selectfont is executed twice but I don't care about speed...
599         \expandafter\g@addto@macro\csname rmfamily \endcsname
600             {\kanjifamily\mcdefault\selectfont}%
601         \expandafter\g@addto@macro\csname sffamily \endcsname
602             {\kanjifamily\gtdefault\selectfont}%
603         \expandafter\g@addto@macro\csname ttfamily \endcsname
604             {\kanjifamily\gtdefault\selectfont}%
605     \fi}
606 \else % 2020-02-02
607 \g@addto@macro\@rmfamilyhook
608     {\prepare@family@series@update@kanji{mc}\mcdefault}
609 \g@addto@macro\@sffamilyhook
610     {\prepare@family@series@update@kanji{gt}\gtdefault}
611 \g@addto@macro\@ttfamilyhook
612     {\prepare@family@series@update@kanji{gt}\gtdefault}
613 \fi
614 \else % --- for 2020-02-02 or older END & for 2020-10-01 BEGIN
615 \AddToHook{rmfamily}%
616     {\prepare@family@series@update@kanji{mc}\mcdefault}
617 \AddToHook{sffamily}%
618     {\prepare@family@series@update@kanji{gt}\gtdefault}
619 \AddToHook{ttfamily}%
620     {\prepare@family@series@update@kanji{gt}\gtdefault}
621 \fi % --- for 2020-10-01 END
622 </class | minijs>
623 <*class>

```

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

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

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

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

```
624 \ifx\DeclareFixJFMCJKTextFontCommand\@undefined
625 \DeclareRobustCommand\textmc[1]{%
626   \relax\ifmmode \expandafter\nfss@text \fi{\mcfamily #1}}
627 \DeclareRobustCommand\textgt[1]{%
628   \relax\ifmmode \expandafter\nfss@text \fi{\gtfamily #1}}
629 \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。

```
630 \def\reDeclareMathAlphabet#1#2#3{%
631   \edef\@tempa{\expandafter\@gobble\string#2}%
632   \edef\@tempb{\expandafter\@gobble\string#3}%
633   \edef\@tempc{\string \@expandafter\@gobbletwo\string#2}%
634   \ifx\@tempc\@tempa%
635     \edef\@tempa{\expandafter\@gobbletwo\string#2}%
636     \edef\@tempb{\expandafter\@gobbletwo\string#3}%
637   \fi
638   \begingroup
639     \let\protect\noexpand
640     \def\@tempaa{\relax}%
641     \expandafter\ifx\csname RDMAorg@\@tempa\endcsname\relax
642       \edef\@tempaa{\expandafter\def\expandafter\noexpand%
643         \csname RDMAorg@\@tempa\endcsname{%
644           \expandafter\noexpand\csname\@tempa\endcsname}}%
645     \fi
646     \def\@tempbb{\relax}%
647     \expandafter\ifx\csname RDMAorg@\@tempb\endcsname\relax
648       \edef\@tempbb{\expandafter\def\expandafter\noexpand%
649         \csname RDMAorg@\@tempb\endcsname{%
650           \expandafter\noexpand\csname\@tempb\endcsname}}%
651     \fi
652     \edef\@tempc{\@tempaa\@tempbb}%
653   \expandafter\endgroup\@tempc%
654   \edef#1{\noexpand\protect\expandafter\noexpand\csname%
655     \expandafter\@gobble\string#1\space\space\endcsname}%

```

```

656 \expandafter\edef\csname\expandafter\@gobble\string#1\space\space\endcsname%
657 {\noexpand\DualLang@mathalph@bet%
658 {\expandafter\noexpand\csname RDMAorg@\@tempa\endcsname}%
659 {\expandafter\noexpand\csname RDMAorg@\@tempb\endcsname}%
660 }%
661 }
662 \@onlypreamble\reDeclareMathAlphabet
663 \def\DualLang@mathalph@bet#1#2{%
664 \relax\ifmmode
665 \ifx\math@bgroup\bgroup% 2e normal style (\mathrm{...})
666 \bgroup\let\DualLang@Mfontsw\DLMfontsw@standard
667 \else
668 \ifx\math@bgroup\relax% 2e two letter style (\rm->\mathrm)
669 \let\DualLang@Mfontsw\DLMfontsw@oldstyle
670 \else
671 \ifx\math@bgroup\@empty% 2.09 oldfont style ({\mathrm ...})
672 \let\DualLang@Mfontsw\DLMfontsw@oldfont
673 \else% panic! assume 2e normal style
674 \bgroup\let\DualLang@Mfontsw\DLMfontsw@standard
675 \fi
676 \fi
677 \fi
678 \else
679 \let\DualLang@Mfontsw\@firstoftwo
680 \fi
681 \DualLang@Mfontsw{#1}{#2}%
682 }
683 \def\DLMfontsw@standard#1#2#3{#1{#2{#3}}\egroup}
684 \def\DLMfontsw@oldstyle#1#2{#1\relax\@fontswitch\relax{#2}}
685 \def\DLMfontsw@oldfont#1#2{#1\relax#2\relax}
686 \if@enablejfam
687 \DeclareSymbolFont{mincho}{\jsc@JYn}{mc}{m}{n}
688 \DeclareSymbolFontAlphabet{\mathmc}{mincho}
689 \SetSymbolFont{mincho}{bold}{\jsc@JYn}{gt}{m}{n}
690 \jfam\symmincho
691 \DeclareMathAlphabet{\mathgt}{\jsc@JYn}{gt}{m}{n}
692 \AtBeginDocument{%
693 \reDeclareMathAlphabet{\mathrm}{\@mathrm}{\@mathmc}
694 \reDeclareMathAlphabet{\mathbf}{\@mathbf}{\@mathgt}}
695 \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。

```
696 % \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。

```
697 \prebreakpenalty\jis"2147=10000      % 5000      '
698 \postbreakpenalty\jis"2148=10000      % 5000      "
699 \prebreakpenalty\jis"2149=10000      % 5000      "
```

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

```
700 \inhibitxspcode'!=1
701 \inhibitxspcode'\pi=2
```

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

```
702 % \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。

```
703 \xspcode' +=3
704 \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。

```
705 \xspcode'^^80=3
706 \xspcode'^^81=3
707 \xspcode'^^82=3
708 \xspcode'^^83=3
709 \xspcode'^^84=3
710 \xspcode'^^85=3
711 \xspcode'^^86=3
712 \xspcode'^^87=3
713 \xspcode'^^88=3
714 \xspcode'^^89=3
715 \xspcode'^^8a=3
716 \xspcode'^^8b=3
717 \xspcode'^^8c=3
718 \xspcode'^^8d=3
719 \xspcode'^^8e=3
720 \xspcode'^^8f=3
721 \xspcode'^^90=3
722 \xspcode'^^91=3
723 \xspcode'^^92=3
724 \xspcode'^^93=3
```

```

725 \xspcode'^94=3
726 \xspcode'^95=3
727 \xspcode'^96=3
728 \xspcode'^97=3
729 \xspcode'^98=3
730 \xspcode'^99=3
731 \xspcode'^9a=3
732 \xspcode'^9b=3
733 \xspcode'^9c=3
734 \xspcode'^9d=3
735 \xspcode'^9e=3
736 \xspcode'^9f=3
737 \xspcode'^a0=3
738 \xspcode'^a1=3
739 \xspcode'^a2=3
740 \xspcode'^a3=3
741 \xspcode'^a4=3
742 \xspcode'^a5=3
743 \xspcode'^a6=3
744 \xspcode'^a7=3
745 \xspcode'^a8=3
746 \xspcode'^a9=3
747 \xspcode'^aa=3
748 \xspcode'^ab=3
749 \xspcode'^ac=3
750 \xspcode'^ad=3
751 \xspcode'^ae=3
752 \xspcode'^af=3
753 \xspcode'^b0=3
754 \xspcode'^b1=3
755 \xspcode'^b2=3
756 \xspcode'^b3=3
757 \xspcode'^b4=3
758 \xspcode'^b5=3
759 \xspcode'^b6=3
760 \xspcode'^b7=3
761 \xspcode'^b8=3
762 \xspcode'^b9=3
763 \xspcode'^ba=3
764 \xspcode'^bb=3
765 \xspcode'^bc=3
766 \xspcode'^bd=3
767 \xspcode'^be=3
768 \xspcode'^bf=3
769 \xspcode'^c0=3
770 \xspcode'^c1=3
771 \xspcode'^c2=3
772 \xspcode'^c3=3
773 \xspcode'^c4=3

```


774 \xspcode'^^c5=3
775 \xspcode'^^c6=3
776 \xspcode'^^c7=3
777 \xspcode'^^c8=3
778 \xspcode'^^c9=3
779 \xspcode'^^ca=3
780 \xspcode'^^cb=3
781 \xspcode'^^cc=3
782 \xspcode'^^cd=3
783 \xspcode'^^ce=3
784 \xspcode'^^cf=3
785 \xspcode'^^d0=3
786 \xspcode'^^d1=3
787 \xspcode'^^d2=3
788 \xspcode'^^d3=3
789 \xspcode'^^d4=3
790 \xspcode'^^d5=3
791 \xspcode'^^d6=3
792 \xspcode'^^d7=3
793 \xspcode'^^d8=3
794 \xspcode'^^d9=3
795 \xspcode'^^da=3
796 \xspcode'^^db=3
797 \xspcode'^^dc=3
798 \xspcode'^^dd=3
799 \xspcode'^^de=3
800 \xspcode'^^df=3
801 \xspcode'^^e0=3
802 \xspcode'^^e1=3
803 \xspcode'^^e2=3
804 \xspcode'^^e3=3
805 \xspcode'^^e4=3
806 \xspcode'^^e5=3
807 \xspcode'^^e6=3
808 \xspcode'^^e7=3
809 \xspcode'^^e8=3
810 \xspcode'^^e9=3
811 \xspcode'^^ea=3
812 \xspcode'^^eb=3
813 \xspcode'^^ec=3
814 \xspcode'^^ed=3
815 \xspcode'^^ee=3
816 \xspcode'^^ef=3
817 \xspcode'^^f0=3
818 \xspcode'^^f1=3
819 \xspcode'^^f2=3
820 \xspcode'^^f3=3
821 \xspcode'^^f4=3
822 \xspcode'^^f5=3

```

823 \xspcode'^^f6=3
824 \xspcode'^^f7=3
825 \xspcode'^^f8=3
826 \xspcode'^^f9=3
827 \xspcode'^^fa=3
828 \xspcode'^^fb=3
829 \xspcode'^^fc=3
830 \xspcode'^^fd=3
831 \xspcode'^^fe=3
832 \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。

```

833 \def\@{\spacefactor3000{}}

```

4 フォントサイズ

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

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

のようにして行います。これは

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

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

<code>\@vpt</code>	5	<code>\@vipt</code>	6	<code>\@viipt</code>	7
<code>\@viiipt</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 \LaTeX 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ものは、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。

```

834 </class>
835 <*class | minijs>
836 %% \@setfontsize with \parindent and \(\x)kanjiskip settings
837 \def\@setfontsize#1#2#3{%
838 <minijs> \nomath#1%
839 \ifx\protect\@typeset@protect
840 \let\@currsz#1%
841 \fi
842 \fontsize{#2}{#3}\selectfont
843 \ifdim\parindent>\z@
844 <class> \if@english
845 <class> \parindent=1em
846 <class> \else
847 \parindent=1zw
848 <class> \fi
849 \fi
850 \kanjiskip=0zw plus .1zw minus .01zw
851 <class> \ifdim\xkanjiskip>\z@
852 <class> \if@slide \xkanjiskip=0.1em \else
853 \xkanjiskip=0.25em plus 0.15em minus 0.06em
854 <class> \fi
855 <class> \fi
856 }
857 </class | minijs>
858 <*class>

```

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

```

859 \def\jsc@setfontsize#1#2#3{%
860 \@setfontsize#1{#2\jsc@empt}{#3\jsc@empt}}

```

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

```

861 \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。

```

862 \newif\ifnarrowbaselines
863 \if@english
864   \narrowbaselinestrue
865 \fi
866 \def\narrowbaselines{%
867   \narrowbaselinestrue
868   \skip0=\abovedisplayskip
869   \skip2=\abovedisplayshortskip
870   \skip4=\belowdisplayskip
871   \skip6=\belowdisplayshortskip
872   \@currsize\selectfont
873   \abovedisplayskip=\skip0
874   \abovedisplayshortskip=\skip2
875   \belowdisplayskip=\skip4
876   \belowdisplayshortskip=\skip6\relax}
877 \def\widebaselines{\narrowbaselinesfalse\@currsize\selectfont}

```

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

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

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

```

878 \renewcommand{\normalsize}{%
879   \ifnarrowbaselines
880     \jsc@setfontsize\normalsize\@xpt\@xipt
881   \else
882     \jsc@setfontsize\normalsize\@xpt{\n@baseline}%
883   \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] $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ Q & A 52569 から始まる議論について逡巡していましたが、結局、微調節してみることにしました。

```

884   \abovedisplayskip 11\jsc@mpt \@plus3\jsc@mpt \@minus4\jsc@mpt
885   \abovedisplayshortskip \z@ \@plus3\jsc@mpt
886   \belowdisplayskip 9\jsc@mpt \@plus3\jsc@mpt \@minus4\jsc@mpt
887   \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。

```

888   \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。

```

889 \end{class}

```

```

890 <*class | minijs>
891 %% initialize
892 \normalsize
893 </class | minijs>
894 <*class>

```

\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
\Cdp ラメータに実際の値を設定します。たとえば \Cwd は \normalfont の全角幅 (1zw) です。
\Cwd [2017-08-31] 基準とする文字を「全角空白」(EUC コード 0xA1A1) から「漢」(JIS コー
\Cvs ド 0x3441) へ変更しました。

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

```

895 \setbox0\hbox{\char\jis"3441}%
896 \setlength\Cht{\ht0}
897 \setlength\Cdp{\dp0}
898 \setlength\Cwd{\wd0}
899 \setlength\Cvs{\baselineskip}
900 \setlength\Chs{\wd0}
901 \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。

```

902 \newcommand{\small}{%
903   \ifnarrowbaselines
904     <|kiyou>   \jsc@setfontsize\small\@ixpt{11}%
905     <kiyou>    \jsc@setfontsize\small{8.8888}{11}%
906   \else
907     <|kiyou>   \jsc@setfontsize\small\@ixpt{13}%
908     <kiyou>    \jsc@setfontsize\small{8.8888}{13.2418}%
909   \fi
910   \abovedisplayskip 9\jsc@mpt \@plus3\jsc@mpt \@minus4\jsc@mpt
911   \abovedisplayshortskip \z@ \@plus3\jsc@mpt
912   \belowdisplayskip \abovedisplayskip
913   \belowdisplayshortskip \belowdisplayskip
914   \def\@listi{\leftmargin\leftmargini
915     \topsep \z@
916     \parsep \z@
917     \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。

```

918 \newcommand{\footnotesize}{%
919   \ifnarrowbaselines
920     <|kiyou>   \jsc@setfontsize\footnotesize\@viipt{9.5}%
921     <kiyou>    \jsc@setfontsize\footnotesize{8.8888}{11}%
922   \else

```

```

923 <|kiyou> \jsc@setfontsize\footnotesize\@viipt{11}%
924 <kiyou> \jsc@setfontsize\footnotesize{8.8888}{13.2418}%
925 \fi
926 \abovedisplayskip 6\jsc@mpt \@plus2\jsc@mpt \@minus3\jsc@mpt
927 \abovedisplayshortskip \z@ \@plus2\jsc@mpt
928 \belowdisplayskip \abovedisplayskip
929 \belowdisplayshortskip \belowdisplayskip
930 \def\@listif\leftmargin\leftmargin
931 \topsep \z@
932 \parsep \z@
933 \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 934 \newcommand{\scriptsize}{\jsc@setfontsize\scriptsize\@viipt\@viipt}
935 \newcommand{\tiny}{\jsc@setfontsize\tiny\@vpt\@vpt}
\Huge 936 \if@twocolumn
\HUGE 937 <|kiyou> \newcommand{\large}{\jsc@setfontsize\large\@xiipt{\n@baseline}}
938 <kiyou> \newcommand{\large}{\jsc@setfontsize\large{11.111}{\n@baseline}}
939 \else
940 <|kiyou> \newcommand{\large}{\jsc@setfontsize\large\@xiipt{17}}
941 <kiyou> \newcommand{\large}{\jsc@setfontsize\large{11.111}{17}}
942 \fi
943 <|kiyou> \newcommand{\Large}{\jsc@setfontsize\Large\@xivpt{21}}
944 <kiyou> \newcommand{\Large}{\jsc@setfontsize\Large{12.222}{21}}
945 \newcommand{\LARGE}{\jsc@setfontsize\LARGE\@xviipt{25}}
946 \newcommand{\huge}{\jsc@setfontsize\huge\@xxpt{28}}
947 \newcommand{\Huge}{\jsc@setfontsize\Huge\@xxvpt{33}}
948 \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。

```

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

```

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

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

```
950 % \newcommand{\headfont}{\bfseries}
951 \newcommand{\headfont}{\gtfamily\sffamily}
952 % \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。

```
953 <|kiyou>\setlength\columnsep{2zw}
954 <|kiyou>\setlength\columnsep{28truebp}
955 \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 956 \setlength\lineskip{1\jsc@mp}
957 \setlength\normallineskip{1\jsc@mp}
\normallineskiplimit 958 \setlength\lineskiplimit{1\jsc@mp}
959 \setlength\normallineskiplimit{1\jsc@mp}
```

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

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

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

```
960 \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。

```
961 \setlength\parskip{\z@}
962 \if@slide
963   \setlength\parindent{0zw}
964 \else
965   \setlength\parindent{1zw}
966 \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 967 \@lowpenalty 51
```

```
968 \@medpenalty 151
969 \@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。

```
970 % \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。

```
971 % \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。

```
972 \setlength\topskip{1.38zw}%% from 10\jsc@mpt (2016-08-17)
973 \if@slide
974   \setlength\headheight{0\jsc@mpt}
975 \else
976   \setlength\headheight{20\jsc@mpt}%% from 2\topskip (2016-08-17); from \topskip (2003-06-26)
977 \fi
```

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

```
978 <*article | kiyou>
979 \if@slide
980   \setlength\footskip{0pt}
981 \else
982   \setlength\footskip{0.03367\paperheight}
983   \ifdim\footskip<\baselineskip
984     \setlength\footskip{\baselineskip}
985   \fi
986 \fi
987 </article | kiyou>
988 <jspf>\setlength\footskip{9\jsc@mmm}
989 <*book>
990 \if@report
```



```

991 \setlength\footskip{0.03367\paperheight}
992 \ifdim\footskip<\baselineskip
993   \setlength\footskip{\baselineskip}
994 \fi
995 \else
996   \setlength\footskip{0pt}
997 \fi
998 </book>
999 <*report>
1000 \setlength\footskip{0.03367\paperheight}
1001 \ifdim\footskip<\baselineskip
1002   \setlength\footskip{\baselineskip}
1003 \fi
1004 </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)。

```

1005 <*article>
1006 \if@slide
1007   \setlength\headsep{0\jsc@empt}
1008   \addtolength\headsep{-\topskip}%% added (2016-10-08)
1009   \addtolength\headsep{10\jsc@empt}%% added (2016-10-08)
1010 \else
1011   \setlength\headsep{\footskip}
1012   \addtolength\headsep{-\topskip}
1013 \fi
1014 </article>
1015 <*book>
1016 \if@report
1017   \setlength\headsep{\footskip}
1018   \addtolength\headsep{-\topskip}
1019 \else
1020   \setlength\headsep{6\jsc@mmm}
1021   \addtolength\headsep{-\topskip}%% added (2016-10-08)
1022   \addtolength\headsep{10\jsc@empt}%% added (2016-10-08)
1023 \fi
1024 </book>
1025 <*report>
1026 \setlength\headsep{\footskip}
1027 \addtolength\headsep{-\topskip}
1028 </report>
1029 <*jspf>
1030 \setlength\headsep{9\jsc@mmm}
1031 \addtolength\headsep{-\topskip}
1032 </jspf>

```

```

1033 <*kiyou>
1034 \setlength\headheight{0\jsc@empt}
1035 \setlength\headsep{0\jsc@empt}
1036 \addtolength\headsep{-\topskip}%% added (2016-10-08)
1037 \addtolength\headsep{10\jsc@empt}%% added (2016-10-08)
1038 </kiyou>

```

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

```

1039 \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。

```

1040 \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。

```

1041 <*article>
1042 \if@slide
1043   \setlength\fullwidth{0.9\paperwidth}
1044 \else
1045   \setlength\fullwidth{0.76\paperwidth}
1046 \fi
1047 \if@twocolumn \@tempdima=2zw \else \@tempdima=1zw \fi
1048 \divide\fullwidth\@tempdima \multiply\fullwidth\@tempdima
1049 \setlength\textwidth{\fullwidth}
1050 </article>
1051 <*book>
1052 \if@report
1053   \setlength\fullwidth{0.76\paperwidth}
1054 \else
1055   \setlength\fullwidth{\paperwidth}
1056   \addtolength\fullwidth{-36\jsc@mmm}
1057 \fi
1058 \if@twocolumn \@tempdima=2zw \else \@tempdima=1zw \fi

```

```

1059 \divide\fullwidth\@tempdima \multiply\fullwidth\@tempdima
1060 \setlength\textwidth{\fullwidth}
1061 \if@report \else
1062   \if@twocolumn \else
1063     \ifdim \fullwidth>40zw
1064       \setlength\textwidth{40zw}
1065     \fi
1066   \fi
1067 \fi
1068 \</book>
1069 \<*report>
1070 \setlength\fullwidth{0.76\paperwidth}
1071 \if@twocolumn \@tempdima=2zw \else \@tempdima=1zw \fi
1072 \divide\fullwidth\@tempdima \multiply\fullwidth\@tempdima
1073 \setlength\textwidth{\fullwidth}
1074 \</report>
1075 \<*jpf>
1076 \setlength\fullwidth{50zw}
1077 \addtolength\fullwidth{8\jsc@mmm}
1078 \setlength\textwidth{\fullwidth}
1079 \</jpf>
1080 \<*kiyou>
1081 \setlength\fullwidth{48zw}
1082 \addtolength\fullwidth{\columnsep}
1083 \setlength\textwidth{\fullwidth}
1084 \</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つたので、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)。

```

1085 \<*article | book | report>
1086 \if@slide
1087   \setlength{\textheight}{0.95\paperheight}
1088 \else
1089   \setlength{\textheight}{0.83\paperheight}

```

```

1090 \fi
1091 \addtolength{\textheight}{-10\jsc@empt}%% from -\topskip (2016-10-08); from -\headheight (200
1092 \addtolength{\textheight}{-\headsep}
1093 \addtolength{\textheight}{-\footskip}
1094 \addtolength{\textheight}{-\topskip}
1095 \divide\textheight\baselineskip
1096 \multiply\textheight\baselineskip
1097 </article | book | report>
1098 <jspf>\setlength{\textheight}{51\baselineskip}
1099 <kiyou>\setlength{\textheight}{47\baselineskip}
1100 \addtolength{\textheight}{\topskip}
1101 \addtolength{\textheight}{0.1\jsc@empt}
1102 <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。

```

1103 \def\flushbottom{%
1104 \def\@textbottom{\vskip \z@ \@plus.1\jsc@empt}%
1105 \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。

```

1106 \setlength\marginparsep{\columnsep}
1107 \setlength\marginparpush{\baselineskip}

```

`\oddsidemargin` それぞれ奇数ページ、偶数ページの左マージンから 1 インチ引いた値です。片面印刷では `\evensidemargin` `\oddsidemargin` が使われます。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。

```

1108 \setlength{\oddsidemargin}{\paperwidth}
1109 \addtolength{\oddsidemargin}{-\fullwidth}
1110 \setlength{\oddsidemargin}{.5\oddsidemargin}
1111 \iftombow
1112 \addtolength{\oddsidemargin}{-1in}
1113 \else
1114 \addtolength{\oddsidemargin}{-\inv@mag in}
1115 \fi
1116 \setlength{\evensidemargin}{\oddsidemargin}
1117 \if@mparswitch
1118 \addtolength{\evensidemargin}{\fullwidth}
1119 \addtolength{\evensidemargin}{-\textwidth}
1120 \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。

```
1121 \setlength\marginparwidth{\paperwidth}
1122 \addtolength\marginparwidth{-\oddsidemargin}
1123 \addtolength\marginparwidth{-\inv@mag in}
1124 \addtolength\marginparwidth{-\textwidth}
1125 \addtolength\marginparwidth{-10\jsc@mmm}
1126 \addtolength\marginparwidth{-\marginparsep}
1127 \@tempdima=1zw
1128 \divide\marginparwidth\@tempdima
1129 \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)。

```
1130 \setlength\topmargin{\paperheight}
1131 \addtolength\topmargin{-\textheight}
1132 \if@slide
1133   \addtolength\topmargin{-\headheight}
1134 \else
1135   \addtolength\topmargin{-10\jsc@empt}%% from -\topskip (2016-10-08); from -\headheight (2003
1136 \fi
1137 \addtolength\topmargin{-\headsep}
1138 \addtolength\topmargin{-\footskip}
1139 \setlength\topmargin{0.5\topmargin}
1140 <kiyou>\setlength\topmargin{81truebp}
1141 \iftombow
1142   \addtolength\topmargin{-1in}
1143 \else
1144   \addtolength\topmargin{-\inv@mag in}
1145 \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。

```
1146 {\footnotesize\global\setlength\footnotesep{\baselineskip}}
1147 \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。

```
1148 \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 \LaTeX 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。

```
1149 \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。

```
1150 \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。

```
1151 \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。

```
1152 \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。

```
1153 \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。

```
1154 \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。

```
1155 \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。

```
1156 \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。

```
1157 \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。

```
1158 \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。

```

1159 \setlength\floatsep {12\jsc@empt \@plus 2\jsc@empt \@minus 2\jsc@empt}
1160 \setlength\textfloatsep{20\jsc@empt \@plus 2\jsc@empt \@minus 4\jsc@empt}
1161 \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 1162 \setlength\dblfloatsep {12\jsc@empt \@plus 2\jsc@empt \@minus 2\jsc@empt}
1163 \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 1164 \setlength\@fptop{0\jsc@empt \@plus 1fil}
1165 \setlength\@fpsep{8\jsc@empt \@plus 2fil}
1166 \setlength\@fpbot{0\jsc@empt \@plus 1fil}

```

`\@dblfpsep` 段抜きフロートについての値です。

```

\@dblfpsep 1167 \setlength\@dblfpsep{0\jsc@empt \@plus 1fil}
1168 \setlength\@dblfpsep{8\jsc@empt \@plus 2fil}
\@dblfpbot 1169 \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_LT_EX の標準クラス 2017/02/15 に合わせて, 同じ命令を追
`\pltx@cleartoleftpage` 加しました。

```

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

```

となっています。

```

1170 <{*article | book | report}
1171 \def\pltx@cleartorightpage{\clearpage\if@twoside
1172   \ifodd\c@page
1173     \iftdir
1174       \hbox{}\thispagestyle{empty}\newpage
1175       \if@twocolumn\hbox{}\newpage\fi
1176     \fi
1177   \else
1178     \ifydir
1179       \hbox{}\thispagestyle{empty}\newpage
1180       \if@twocolumn\hbox{}\newpage\fi
1181     \fi
1182   \fi\fi}
1183 \def\pltx@cleartoleftpage{\clearpage\if@twoside
1184   \ifodd\c@page
1185     \ifydir
1186       \hbox{}\thispagestyle{empty}\newpage

```

```

1187     \if@twocolumn\hbox{}\newpage\fi
1188     \fi
1189     \else
1190     \iftdir
1191     \hbox{}\thispagestyle{empty}\newpage
1192     \if@twocolumn\hbox{}\newpage\fi
1193     \fi
1194     \fi\fi}
1195 \def\pltx@cleartooddpage{\clearpage\if@twoside
1196 \ifodd\c@page\else
1197     \hbox{}\thispagestyle{empty}\newpage
1198     \if@twocolumn\hbox{}\newpage\fi
1199     \fi\fi}
1200 \def\pltx@cleartoevenpage{\clearpage\if@twoside
1201 \ifodd\c@page
1202     \hbox{}\thispagestyle{empty}\newpage
1203     \if@twocolumn\hbox{}\newpage\fi
1204     \fi\fi}
1205 </article | book | report>

```

`\cleardoublepage` [2017-02-24] コミュニティ版 p \LaTeX の標準クラス 2017/02/15 に合わせて、report と book クラスの場合に `\cleardoublepage` を再定義します。

```

1206 <*book | report>
1207 \if@openleft
1208     \let\cleardoublepage\pltx@cleartoleftpage
1209 \else\if@openright
1210     \let\cleardoublepage\pltx@cleartorightpage
1211 \fi\fi
1212 </book | report>

```

7 ページスタイル

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

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

ページスタイルは `\ps@...` の形のマクロで定義されています。

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

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

`\markboth{左}{右}` 両方の柱を設定します。

<code>\markright{右}</code>	右の柱を設定します。
<code>\leftmark</code>	左の柱を出力します。
<code>\rightmark</code>	右の柱を出力します。

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

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

`\ps@empty empty` ページスタイルの定義です。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。

```
1213 % \def\ps@empty{%
1214 %   \let\@mkboth\@gobbletwo
1215 %   \let\@oddhead\@empty
1216 %   \let\@oddfoot\@empty
1217 %   \let\@evenhead\@empty
1218 %   \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。

```
1219 \def\ps@plainfoot{%
1220   \let\@mkboth\@gobbletwo
1221   \let\@oddhead\@empty
1222   \def\@oddfoot{\normalfont\hfil\thepage\hfil}%
1223   \let\@evenhead\@empty
1224   \let\@evenfoot\@oddfoot}
1225 \def\ps@plainhead{%
1226   \let\@mkboth\@gobbletwo
1227   \let\@oddfoot\@empty
1228   \let\@evenfoot\@empty
1229   \def\@evenhead{%
1230     \if@mparswitch \hss \fi
1231     \hbox to \fullwidth{\textbf{\thepage}\hfil}%
1232     \if@mparswitch\else \hss \fi}%
1233   \def\@oddhead{%
1234     \hbox to \fullwidth{\hfil\textbf{\thepage}}\hss}}
1235 <book>\if@report \let\ps@plain\ps@plainfoot \else \let\ps@plain\ps@plainhead \fi
1236 <!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。

```
1237 <*article | kiyou>
1238 \if@twoside
1239   \def\ps@headings{%
```

```

1240 \let\@oddfoot\@empty
1241 \let\@evenfoot\@empty
1242 \def\@evenhead{\if@mparswitch \hss \fi
1243 \underline{\hbox to \fullwidth{\textbf{\thepage}\hfil\leftmark}}}%
1244 \if@mparswitch\else \hss \fi}%
1245 \def\@oddhead{%
1246 \underline{%
1247 \hbox to \fullwidth{\rightmark}\hfil\textbf{\thepage}}}\hss}%
1248 \let\@mkboth\markboth
1249 \def\sectionmark##1{\markboth{%
1250 \ifnum \c@secnumdepth >\z@ \thesection \hskip1zw\fi
1251 ##1}}}%
1252 \def\subsectionmark##1{\markright{%
1253 \ifnum \c@secnumdepth >\@ne \thesubsection \hskip1zw\fi
1254 ##1}}}%
1255 }
1256 \else % if not twoside
1257 \def\ps@headings{%
1258 \let\@oddfoot\@empty
1259 \def\@oddhead{%
1260 \underline{%
1261 \hbox to \fullwidth{\rightmark}\hfil\textbf{\thepage}}}\hss}%
1262 \let\@mkboth\markboth
1263 \def\sectionmark##1{\markright{%
1264 \ifnum \c@secnumdepth >\z@ \thesection \hskip1zw\fi
1265 ##1}}}%
1266 \fi
1267 </article | kiyou>

```

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

```

1268 <*book | report>
1269 \newif\if@omit@number
1270 \def\ps@headings{%
1271 \let\@oddfoot\@empty
1272 \let\@evenfoot\@empty
1273 \def\@evenhead{%
1274 \if@mparswitch \hss \fi
1275 \underline{\hbox to \fullwidth{\autoxspacing
1276 \textbf{\thepage}\hfil\leftmark}}}%
1277 \if@mparswitch\else \hss \fi}%
1278 \def\@oddhead{\underline{\hbox to \fullwidth{\autoxspacing
1279 {\if@twoside\rightmark\else\leftmark\fi}\hfil\textbf{\thepage}}}\hss}%
1280 \let\@mkboth\markboth
1281 \def\chaptermark##1{\markboth{%
1282 \ifnum \c@secnumdepth >\m@ne
1283 <book> \if@mainmatter
1284 \if@omit@number\else
1285 \@chapapp\thechapter\@chappos\hskip1zw

```

```

1286         \fi
1287 <book>         \fi
1288         \fi
1289         ##1}{}}%
1290 \def\sectionmark##1{\markright{%
1291     \ifnum \c@secnumdepth >\z@ \thesection \hskip1zw\fi
1292     ##1}}}%
1293 </book | report>

    最後は学会誌の場合です。

1294 <*jspf>
1295 \def\ps@headings{%
1296     \def\@oddfoot{\normalfont\hfil\thepage\hfil}
1297     \def\@evenfoot{\normalfont\hfil\thepage\hfil}
1298     \def\@oddhead{\normalfont\hfil \@title \hfil}
1299     \def\@evenhead{\normalfont\hfil プラズマ・核融合学会誌 \hfil}}
1300 </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。

```

1301 \def\ps@myheadings{%
1302     \let\@oddfoot\@empty\let\@evenfoot\@empty
1303     \def\@evenhead{%
1304         \if@mparswitch \hss \fi%
1305         \hbox to \fullwidth{\thepage\hfil\leftmark}%
1306         \if@mparswitch\else \hss \fi}%
1307     \def\@oddhead{%
1308         \hbox to \fullwidth{\rightmark\hfil\thepage}\hss}%
1309     \let\@mkboth\@gobbletwo
1310 <book | report> \let\chaptermark\@gobble
1311 \let\sectionmark\@gobble
1312 <!book&!report> \let\subsectionmark\@gobble
1313 }

```

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 1314 % \newcommand*{\title}[1]{\gdef\@title{#1}}
      1315 % \newcommand*{\author}[1]{\gdef\@author{#1}}
      1316 % \newcommand*{\date}[1]{\gdef\@date{#1}}
      1317 % \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`

`\keywords`

```

1318 <*jspf>
1319 \newcommand*{\etitle}[1]{\gdef\@etitle{#1}}
1320 \newcommand*{\eauthor}[1]{\gdef\@eauthor{#1}}
1321 \newcommand*{\keywords}[1]{\gdef\@keywords{#1}}
1322 \newcommand*{\email}[1]{\gdef\authors@mail{#1}}
1323 \newcommand*{\AuthorsEmail}[1]{\gdef\authors@mail{author's e-mail:\ #1}}
1324 </jspf>

```

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

```

1325 \def\plainifnotempty{%
1326   \ifx \@oddhead \@empty
1327     \ifx \@oddfoot \@empty
1328       \else
1329         \thispagestyle{plainfoot}%
1330       \fi
1331     \else
1332       \thispagestyle{plainhead}%
1333     \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。

```

1334 <*article | book | report | kiyou>
1335 \if@titlepage
1336   \newcommand{\maketitle}{%
1337     \begin{titlepage}%
1338       \let\footnotesize\small
1339       \let\footnoterule\relax
1340       \let\footnote\thanks
1341       \null\vfil
1342       \if@slide
1343         {\footnotesize \@date}%
1344       \begin{center}
1345         \mbox{} \ \ [1zw]
1346         \large
1347         {\maybeblue\hrule height0\jsc@mpt depth2\jsc@mpt\relax}\par
1348         \jsc@smallskip
1349         \@title
1350         \jsc@smallskip
1351         {\maybeblue\hrule height0\jsc@mpt depth2\jsc@mpt\relax}\par
1352         \vfill

```

```

1353         {\small \@author}%
1354     \end{center}
1355 \else
1356     \vskip 60\jsc@mp
1357     \begin{center}%
1358         {\LARGE \@title \par}%
1359         \vskip 3em%
1360         {\large
1361             \lineskip .75em
1362             \begin{tabular}[t]{c}%
1363                 \@author
1364             \end{tabular}\par}%
1365         \vskip 1.5em
1366         {\large \@date \par}%
1367     \end{center}%
1368 \fi
1369 \par
1370 \@thanks\vfil\null
1371 \end{titlepage}%
1372 \setcounter{footnote}{0}%
1373 \global\let\thanks\relax
1374 \global\let\maketitle\relax
1375 \global\let\@thanks\@empty
1376 \global\let\@author\@empty
1377 \global\let\@date\@empty
1378 \global\let\@title\@empty
1379 \global\let\title\relax
1380 \global\let\author\relax
1381 \global\let\date\relax
1382 \global\let\and\relax
1383 }%
1384 \else
1385 \newcommand{\maketitle}{\par
1386     \begin{group
1387         \renewcommand\thefootnote{\@fnsymbol\c@footnote}%
1388         \def\@makefnmark{\rlap{\@textsuperscript{\normalfont\@thefnmark}}}%
1389         \long\def\@makefntext##1{\advance\leftskip 3zw
1390             \parindent 1zw\noindent
1391             \llap{\@textsuperscript{\normalfont\@thefnmark}\hskip0.3zw}##1}%
1392         \if@twocolumn
1393             \ifnum \col@number=\@ne
1394                 \@maketitle
1395             \else
1396                 \twocolumn[\@maketitle]%
1397             \fi
1398         \else
1399             \newpage
1400             \global\@topnum\z@ % Prevents figures from going at top of page.
1401             \@maketitle

```

```

1402     \fi
1403     \plainifnotempty
1404     \@thanks
1405 \endgroup
1406 \setcounter{footnote}{0}%
1407 \global\let\thanks\relax
1408 \global\let\maketitle\relax
1409 \global\let\@thanks\@empty
1410 \global\let\@author\@empty
1411 \global\let\@date\@empty
1412 \global\let\@title\@empty
1413 \global\let\title\relax
1414 \global\let\author\relax
1415 \global\let\date\relax
1416 \global\let\and\relax
1417 }

```

`\@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。

```

1418 \def\@maketitle{%
1419     \newpage\null
1420     \vskip 2em
1421     \begin{center}%
1422         \let\footnote\thanks
1423         {\LARGE \@title \par}%
1424         \vskip 1.5em
1425         {\large
1426             \lineskip .5em
1427             \begin{tabular}[t]{c}%
1428                 \@author
1429             \end{tabular}\par}%
1430         \vskip 1em
1431         {\large \@date}%
1432     \end{center}%
1433     \par\vskip 1.5em
1434 <article | report | kiyou> \ifvoid\@abstractbox\else\centerline{\box\@abstractbox}\vskip1.5em\fi
1435 }
1436 \fi
1437 </article | book | report | kiyou>
1438 <*jpf>
1439 \newcommand{\maketitle}{\par
1440     \begingroup
1441     \renewcommand\thefootnote{\@fnsymbol\c@footnote}%
1442     \def\@makefnmark{\rlap{\@textsuperscript{\normalfont\@thefnmark}}}%
1443     \long\def\@makefntext##1{\advance\leftskip 3zw
1444         \parindent 1zw\noindent
1445         \llap{\@textsuperscript{\normalfont\@thefnmark}\hskip0.3zw}##1}%
1446     \twocolumn[\@maketitle]%
1447     \plainifnotempty
1448     \@thanks

```

```

1449 \endgroup
1450 \setcounter{footnote}{0}%
1451 \global\let\thanks\relax
1452 \global\let\maketitle\relax
1453 \global\let\@thanks\@empty
1454 \global\let\@author\@empty
1455 \global\let\@date\@empty
1456 % \global\let\@title\@empty % \@title は柱に使う
1457 \global\let\title\relax
1458 \global\let\author\relax
1459 \global\let\date\relax
1460 \global\let\and\relax
1461 \ifx\authors@mail\@undefined\else{%
1462     \def\@makefntext{\advance\leftskip 3zw \parindent -3zw}%
1463     \footnotetext[0]{\itshape\authors@mail}%
1464 } \fi
1465 \global\let\authors@mail\@undefined}
1466 \def\@maketitle{%
1467     \newpage\null
1468     \vskip 6em % used to be 2em
1469     \begin{center}
1470         \let\footnote\thanks
1471         \ifx\@title\@undefined\else{\LARGE\headfont\@title\par}\fi
1472         \lineskip .5em
1473         \ifx\@author\@undefined\else
1474             \vskip 1em
1475             \begin{tabular}[t]{c}%
1476                 \@author
1477             \end{tabular}\par
1478         \fi
1479         \ifx\@etitle\@undefined\else
1480             \vskip 1em
1481             {\large \@etitle \par}%
1482         \fi
1483         \ifx\@eauthor\@undefined\else
1484             \vskip 1em
1485             \begin{tabular}[t]{c}%
1486                 \@eauthor
1487             \end{tabular}\par
1488         \fi
1489         \vskip 1em
1490         \@date
1491     \end{center}
1492     \vskip 1.5em
1493     \centerline{\box\@abstractbox}
1494     \ifx\@keywords\@undefined\else
1495         \vskip 1.5em
1496         \centerline{\parbox{157\jsc@mmm}{\textsf{Keywords:}}\ \small\@keywords}}
1497     \fi

```

```
1498 \vskip 1.5em}
1499 </jpf>
```

8.2 章・節

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

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

それぞれの引数の意味は次の通りです。

名 ユーザレベルコマンドの名前です (例: section)。

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

字下げ 見出しの字下げ量です。

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

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

スタイル 見出しの文字スタイルの設定です。

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

別見出し 目次や柱に出力する見出しです。

見出し 見出しです。

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

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

```
1500 \def\startsection#1#2#3#4#5#6{%
1501 \if@noskipsec \leavevmode \fi
1502 \par
1503 % 見出し上の空きを \@tempskipa にセットする
1504 \@tempskipa #4\relax
1505 % \@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
1506 \if@english \@afterindentfalse \else \@afterindenttrue \fi
1507 % 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
1508 \ifdim \@tempskipa <\z@
1509 \@tempskipa -\@tempskipa \@afterindentfalse
1510 \fi
1511 \if@nobreak
1512 % \everypar{\everyparhook}% これは間違い
1513 \everypar{}%
1514 \else
```



```

1515 \addpenalty\@secpenalty
1516 % 次の行は削除
1517 % \addvspace\@tempskipa
1518 % 次の \noindent まで追加
1519 \ifdim \@tempskipa >\z@
1520 \if@slide\else
1521 \null
1522 \vspace*{-\baselineskip}%
1523 \fi
1524 \vskip\@tempskipa
1525 \fi
1526 \fi
1527 \noindent
1528 % 追加終わり
1529 \@ifstar
1530 {\@ssect{#3}{#4}{#5}{#6}}%
1531 {\@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。

1532 \def\@sect#1#2#3#4#5#6[#7]#8{%
1533 \ifnum #2>\c@secnumdepth
1534 \let\@svsec\@empty
1535 \else
1536 \refstepcounter{#1}%
1537 \protected@edef\@svsec{\@secntformat{#1}\relax}%
1538 \fi
1539 % 見出し後の空きを \@tempskipa にセット
1540 \@tempskipa #5\relax
1541 % 条件判断の順序を入れ替えました
1542 \ifdim \@tempskipa<\z@
1543 \def\@svsechd{%
1544 #6{\hskip #3\relax
1545 \@svsec #8}%
1546 \csname #1mark\endcsname{#7}%
1547 \addcontentsline{toc}{#1}{%
1548 \ifnum #2>\c@secnumdepth \else
1549 \protect\numberline{\csname the#1\endcsname}%
1550 \fi
1551 #7}}% 目次にフルネームを載せるなら #8
1552 \else
1553 \begingroup
1554 \interlinepenalty \@M % 下から移動
1555 #6{%
1556 \@hangfrom{\hskip #3\relax\@svsec}%
1557 % \interlinepenalty \@M % 上に移動
1558 #8\@par}%
1559 \endgroup
1560 \csname #1mark\endcsname{#7}%

```

```

1561 \addcontentsline{toc}{#1}{%
1562 \ifnum #2>\c@secnumdepth \else
1563 \protect\numberline{\csname the#1\endcsname}%
1564 \fi
1565 #7}% 目次にフルネームを載せるならここは #8
1566 \fi
1567 \@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。

```

1568 \def\@xsect#1{%
1569 % 見出しの後ろの空きを \@tempskipa にセット
1570 \@tempskipa #1\relax
1571 % 条件判断の順序を変えました
1572 \ifdim \@tempskipa<\z@
1573 \nobreakfalse
1574 \global\@noskipsecttrue
1575 \everypar{%
1576 \if@noskipsec
1577 \global\@noskipsecfalse
1578 {\setbox\z@\lastbox}%
1579 \clubpenalty\@M
1580 \begingroup \@svsechd \endgroup
1581 \unskip
1582 \@tempskipa #1\relax
1583 \hskip -\@tempskipa
1584 \else
1585 \clubpenalty \@clubpenalty
1586 \everypar{\everyparhook}%
1587 \fi\everyparhook}%
1588 \else
1589 \par \nobreak
1590 \vskip \@tempskipa
1591 \@afterheading
1592 \fi
1593 \if@slide
1594 {\vskip\if@twocolumn-5\jsc@mpt\else-6\jsc@mpt\fi
1595 \maybeblue\hrule height0\jsc@mpt depth1\jsc@mpt
1596 \vskip\if@twocolumn 4\jsc@mpt\else 7\jsc@mpt\fi\relax}%
1597 \fi
1598 \par % 2000-12-18
1599 \ignorespaces}
1600 \def\@ssect#1#2#3#4#5{%
1601 \@tempskipa #3\relax
1602 \ifdim \@tempskipa<\z@
1603 \def\@svsechd{#4{\hskip #1\relax #5}}%

```

```

1604 \else
1605 \begingroup
1606 #4{%
1607 \@hangfrom{\hskip #1}%
1608 \interlinepenalty \@M #5\@@par}%
1609 \endgroup
1610 \fi
1611 \@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 1612 \newcommand*\chaptermark[1]{}
\subsubsectionmark 1613 % \newcommand*\sectionmark[1]{}
1614 % \newcommand*\subsectionmark[1]{}
\paragraphmark 1615 % \newcommand*\subsubsectionmark[1]{}
\subparagraphmark 1616 % \newcommand*\paragraphmark[1]{}
1617 % \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。

```

1618 <|book&!report>\setcounter{secnumdepth}{3}
1619 <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 1620 \newcounter{part}
1621 <book | report>\newcounter{chapter}
\c@subsubsection 1622 <book | report>\newcounter{section}[chapter]
1623 <|book&!report>\newcounter{section}
\c@paragraph 1624 \newcounter{subsection}[section]
1625 \newcounter{subsubsection}[subsection]
1626 \newcounter{paragraph}[subsubsection]
1627 \newcounter{subparagraph}[paragraph]

```

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

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

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

以下ではスペース節約のため @ の付いた内部表現を多用しています。

```
1628 \renewcommand{\thepart}{\@Roman\c@part}
1629 \langle!book&!report\rangle \renewcommand{\thesection}{\@arabic\c@section}
1630 \langle!book&!report\rangle \renewcommand{\thesection}{\presectionname\@arabic\c@section\postsectionname}
1631 \langle!book&!report\rangle \renewcommand{\thesubsection}{\@arabic\c@section.\@arabic\c@subsection}
1632 \langle*book | report\rangle
1633 \renewcommand{\thechapter}{\@arabic\c@chapter}
1634 \renewcommand{\thesection}{\thechapter.\@arabic\c@section}
1635 \renewcommand{\thesubsection}{\thesection.\@arabic\c@subsection}
1636 \langle/book | report\rangle
1637 \renewcommand{\thesubsubsection}{\%
1638   \thesubsection.\@arabic\c@subsubsection}
1639 \renewcommand{\theparagraph}{\%
1640   \thesubsubsection.\@arabic\c@paragraph}
1641 \renewcommand{\thesubparagraph}{\%
1642   \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。

```
1643 \langle book | report \rangle \newcommand{\@chapapp}{\prechaptername}
1644 \langle book | report \rangle \newcommand{\@chappos}{\postchaptername}
```

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

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

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

```
1645 \langle*book\rangle
1646 \newcommand\frontmatter{\%
1647   \pltx@cleartooddpage
1648   \@mainmatterfalse
1649   \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。

```
1650 \newcommand\mainmatter{\%
1651   \pltx@cleartooddpage
1652   \@mainmattertrue
1653   \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。

```

1654 \newcommand\backmatter{%
1655   \if@openleft
1656     \cleardoublepage
1657   \else\if@openright
1658     \cleardoublepage
1659   \else
1660     \clearpage
1661   \fi\fi
1662   \@mainmatterfalse}
1663 \</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。

```

1664 <!*book&!report>
1665 \newcommand\part{%
1666   \if@noskipsec \leavevmode \fi
1667   \par
1668   \addvspace{4ex}%
1669   \if@english \@afterindentfalse \else \@afterindenttrue \fi
1670   \secdef\@part\@spart}
1671 </!book&!report>

```

`book` および `report` クラスの場合は、少し複雑です。

```

1672 <*book | report>
1673 \newcommand\part{%
1674   \if@openleft
1675     \cleardoublepage
1676   \else\if@openright
1677     \cleardoublepage
1678   \else
1679     \clearpage
1680   \fi\fi
1681   \thispagestyle{empty}% 欧文用標準スタイルでは plain
1682   \if@twocolumn
1683     \onecolumn

```

```

1684 \restonecoltrue
1685 \else
1686 \restonecolfalse
1687 \fi
1688 \null\vfil
1689 \secdef\@part\@spart}
1690 </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。

```

1691 < *!book&!report>
1692 \def\@part[#1]#2{%
1693 \ifnum \c@secnumdepth >\m@ne
1694 \refstepcounter{part}%
1695 \addcontentsline{toc}{part}{%
1696 \prepartname\thepart\postpartname\hspace{1zw}#1}%
1697 \else
1698 \addcontentsline{toc}{part}{#1}%
1699 \fi
1700 \markboth{}{}%
1701 {\parindent\z@
1702 \raggedright
1703 \interlinepenalty \@M
1704 \normalfont
1705 \ifnum \c@secnumdepth >\m@ne
1706 \Large\headfont\prepartname\thepart\postpartname
1707 \par\nobreak
1708 \fi
1709 \huge \headfont #2%
1710 \markboth{}{}\par}%
1711 \nobreak
1712 \vskip 3ex
1713 \@afterheading}
1714 < /!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。

```

1715 < *book | report>
1716 \def\@part[#1]#2{%
1717 \ifnum \c@secnumdepth >-2\relax
1718 \refstepcounter{part}%
1719 \addcontentsline{toc}{part}{%
1720 \prepartname\thepart\postpartname\hspace{1zw}#1}%
1721 \else
1722 \addcontentsline{toc}{part}{#1}%
1723 \fi
1724 \markboth{}{}%
1725 {\centering
1726 \interlinepenalty \@M

```

```

1727 \normalfont
1728 \ifnum \c@secnumdepth >-2\relax
1729 \huge\headfont \prepartname\thepart\postpartname
1730 \par\vskip20\jsc@mp
1731 \fi
1732 \Huge \headfont #2\par}%
1733 \@endpart}
1734 </book | report>

```

\@spart 番号を付けない部です。

```

1735 <*\book&!report>
1736 \def\@spart#1{%
1737 \parindent \z@ \raggedright
1738 \interlinepenalty \@M
1739 \normalfont
1740 \huge \headfont #1\par}%
1741 \nobreak
1742 \vskip 3ex
1743 \@afterheading}
1744 <\/!\book&!report>
1745 <*\book | report>
1746 \def\@spart#1{%
1747 \centering
1748 \interlinepenalty \@M
1749 \normalfont
1750 \Huge \headfont #1\par}%
1751 \@endpart}
1752 </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。

```

1753 <*\book | report>
1754 \def\@endpart{\vfil\newpage
1755 \if@twoside
1756 \if@openleft %% added (2017/02/24)
1757 \null\thispagestyle{empty}\newpage
1758 \else\if@openright %% added (2016/12/13)
1759 \null\thispagestyle{empty}\newpage
1760 \fi\fi %% added (2016/12/13, 2017/02/24)
1761 \fi
1762 \if@restonecol
1763 \twocolumn
1764 \fi}
1765 </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。

```

1766 <*book | report>
1767 \newcommand{\chapter}{%
1768   \if@openleft\cleardoublepage\else
1769   \if@openright\cleardoublepage\else\clearpage\fi\fi
1770   \plainifnotempty % 元: \thispagestyle{plain}
1771   \global\@topnum\z@
1772   \if@english \@afterindentfalse \else \@afterindenttrue \fi
1773   \secdef
1774     {\@omit@numberfalse\@chapter}%
1775     {\@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。

```

1776 \def\@chapter[#1]#2{%
1777   \ifnum \c@secnumdepth >\m@ne
1778   <book>   \if@mainmatter
1779             \refstepcounter{chapter}%
1780             \typeout{\@chapapp\thechapter\@chappos}%
1781             \addcontentsline{toc}{chapter}%
1782               {\protect\numberline
1783                % {\if@english\thechapter\else\@chapapp\thechapter\@chappos\fi}%
1784                {\@chapapp\thechapter\@chappos}%
1785                #1}%
1786   <book>   \else\addcontentsline{toc}{chapter}{#1}\fi
1787   \else
1788     \addcontentsline{toc}{chapter}{#1}%
1789   \fi
1790   \chaptermark{#1}%
1791   \addtocontents{lof}{\protect\addvspace{10\jsc@empt}}%
1792   \addtocontents{lot}{\protect\addvspace{10\jsc@empt}}%
1793   \if@twocolumn
1794     \@topnewpage[\@makechapterhead{#2}]%
1795   \else
1796     \@makechapterhead{#2}%
1797     \@afterheading
1798   \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。

```

1799 \def\@makechapterhead#1{%
1800   \vspace*{2\Cvs}% 欧文は 50pt
1801   {\parindent \z@ \raggedright \normalfont
1802    \ifnum \c@secnumdepth >\m@ne
1803   <book>   \if@mainmatter
1804             \huge\headfont \@chapapp\thechapter\@chappos

```



```

1805      \par\nobreak
1806      \vskip \Cvs % 欧文は 20pt
1807 <book>      \fi
1808      \fi
1809      \interlinepenalty\@M
1810      \Huge \headfont #1\par\nobreak
1811      \vskip 3\Cvs}} % 欧文は 40pt

```

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

```

1812 \def\@schapter#1{%
1813   \chaptermark{#1}%
1814   \if@twocolumn
1815     \topnewpage[\@makeschapterhead{#1}]%
1816   \else
1817     \@makeschapterhead{#1}\@afterheading
1818   \fi}

```

\@makeschapterhead 番号なしの章見出しです。

```

1819 \def\@makeschapterhead#1{%
1820   \vspace*{2\Cvs}% 欧文は 50pt
1821   {\parindent \z@ \raggedright
1822     \normalfont
1823     \interlinepenalty\@M
1824     \Huge \headfont #1\par\nobreak
1825     \vskip 3\Cvs}} % 欧文は 40pt
1826 </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。

```

1827 \if@twocolumn
1828   \newcommand{\section}{%
1829     \jpf\ifx\maketitle\relax\else\maketitle\fi
1830     \@startsection{section}{1}{\z@}%
1831     <!kiyou> {0.6\Cvs}{0.4\Cvs}%
1832     <kiyou>  {\Cvs}{0.5\Cvs}%
1833     % {\normalfont\large\headfont\@secapp}}
1834     {\normalfont\large\headfont\raggedright}}
1835 \else
1836   \newcommand{\section}{%
1837     \if@slide\clearpage\fi
1838     \@startsection{section}{1}{\z@}%
1839     {\Cvs \@plus.5\Cdp \@minus.2\Cdp}% 前アキ
1840     {.5\Cvs \@plus.3\Cdp}% 後アキ
1841     % {\normalfont\Large\headfont\@secapp}}
1842     {\normalfont\Large\headfont\raggedright}}

```

1843 \fi

\subsection 同上です。

```
1844 \if@twocolumn
1845   \newcommand{\subsection}{\@startsection{subsection}{2}{\z@}%
1846     {\z@}{\if@slide .4\Cvs \else \z@ \fi}%
1847     {\normalfont\normalsize\headfont}}
1848 \else
1849   \newcommand{\subsection}{\@startsection{subsection}{2}{\z@}%
1850     {\Cvs \@plus.5\Cdp \@minus.2\Cdp}% 前アキ
1851     {.5\Cvs \@plus.3\Cdp}% 後アキ
1852     {\normalfont\large\headfont}}
1853 \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)。

```
1854 \if@twocolumn
1855   \newcommand{\subsubsection}{\@startsection{subsubsection}{3}{\z@}%
1856     {\z@}{\if@slide .4\Cvs \else \z@ \fi}%
1857     {\normalfont\normalsize\headfont}}
1858 \else
1859   \newcommand{\subsubsection}{\@startsection{subsubsection}{3}{\z@}%
1860     {\Cvs \@plus.5\Cdp \@minus.2\Cdp}%
1861     {\if@slide .5\Cvs \@plus.3\Cdp \else \z@ \fi}%
1862     {\normalfont\normalsize\headfont}}
1863 \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。

```
1864 \!jspf\newcommand{\jsParagraphMark}{■}
1865 \if@twocolumn
1866   \newcommand{\paragraph}{\@startsection{paragraph}{4}{\z@}%
1867     {\z@}{\if@slide .4\Cvs \else -1zw\fi}% 改行せず 1zw のアキ
1868 \!jspf} {\normalfont\normalsize\headfont}}
1869 \!jspf} {\normalfont\normalsize\headfont\jsParagraphMark}}
1870 \else
1871   \newcommand{\paragraph}{\@startsection{paragraph}{4}{\z@}%
1872     {0.5\Cvs \@plus.5\Cdp \@minus.2\Cdp}%
1873     {\if@slide .5\Cvs \@plus.3\Cdp \else -1zw\fi}% 改行せず 1zw のアキ
1874 \!jspf} {\normalfont\normalsize\headfont}}
1875 \!jspf} {\normalfont\normalsize\headfont\jsParagraphMark}}
1876 \fi
```

`\subparagraph` 見出しの後ろで改行されません。

```
1877 \if@twocolumn
1878   \newcommand{\subparagraph}{\@startsection{subparagraph}{5}{\z@}%
1879     {\z@}{\if@slide .4\Cvs \@plus.3\Cdp \else -1zw\fi}%
1880     {\normalfont\normalsize\headfont}}
1881 \else
1882   \newcommand{\subparagraph}{\@startsection{subparagraph}{5}{\z@}%
1883     {\z@}{\if@slide .5\Cvs \@plus.3\Cdp \else -1zw\fi}%
1884     {\normalfont\normalsize\headfont}}
1885 \fi
```

8.3 リスト環境

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

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

[2002-05-11] 3zw に変更しました。

[2005-03-19] 二段組は 2zw に戻しました。

```
1886 \if@slide
1887   \setlength\leftmargini{1zw}
1888 \else
1889   \if@twocolumn
1890     \setlength\leftmargini{2zw}
1891   \else
1892     \setlength\leftmargini{3zw}
1893   \fi
1894 \fi
```

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

```
\leftmarginiv 1895 \if@slide
\leftmarginv 1896   \setlength\leftmarginii {1zw}
\leftmarginvi 1897   \setlength\leftmarginiii {1zw}
1898   \setlength\leftmarginiv {1zw}
1899   \setlength\leftmarginv  {1zw}
1900   \setlength\leftmarginvi {1zw}
1901 \else
1902   \setlength\leftmarginii {2zw}
1903   \setlength\leftmarginiii {2zw}
1904   \setlength\leftmarginiv {2zw}
1905   \setlength\leftmarginv  {1zw}
1906   \setlength\leftmarginvi {1zw}
1907 \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。

```
1908 \setlength \labelsep {0.5zw} % .5em
1909 \setlength \labelwidth{\leftmargini}
1910 \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。

```
1911 \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 1912 \@beginparpenalty -\@lowpenalty
\@itempenalty 1913 \@endparpenalty -\@lowpenalty
1914 \@itempenalty -\@lowpenalty
```

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

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

```
1915 \def\@listi{\leftmargin\leftmargini
1916 \parsep \z@
1917 \topsep 0.5\baselineskip
1918 \itemsep \z@ \relax}
1919 \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）。

```
1920 \@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 1921 \def\@listii{\leftmargin\leftmarginii
\@listiv 1922 \labelwidth\leftmarginii \advance\labelwidth-\labelsep
1923 \topsep \z@
\@listv 1924 \parsep \z@
\@listvi 1925 \itemsep\parsep}
1926 \def\@listiii{\leftmargin\leftmarginiii
1927 \labelwidth\leftmarginiii \advance\labelwidth-\labelsep
1928 \topsep \z@
1929 \parsep \z@
1930 \itemsep\parsep}
1931 \def\@listiv {\leftmargin\leftmarginiv
1932 \labelwidth\leftmarginiv
1933 \advance\labelwidth-\labelsep}
1934 \def\@listv {\leftmargin\leftmarginv
1935 \labelwidth\leftmarginv
```

```

1936          \advance\labelwidth-\labelsep}
1937 \def\@listvi {\leftmargin\leftmarginvi
1938          \labelwidth\leftmarginvi
1939          \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。

```

1940 \renewcommand{\theenumi}{\@arabic\c@enumi}
1941 \renewcommand{\theenumii}{\@alph\c@enumii}
1942 \renewcommand{\theenumiii}{\@roman\c@enumiii}
1943 \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 1944 \newcommand{\labelenumi}{\theenumi.}
1945 \newcommand{\labelenumii}{\inhibitglue (\theenumii ) \inhibitglue}
1946 \newcommand{\labelenumiii}{\theenumiii.}
1947 \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 1948 \renewcommand{\p@enumii}{\theenumi}
1949 \renewcommand{\p@enumiii}{\theenumi\inhibitglue (\theenumii )}
1950 \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 1951 \newcommand\labelitemi{\textbullet}
1952 \newcommand\labelitemii{\normalfont\bfseries \textendash}
\labelitemiii 1953 \newcommand\labelitemiii{\textasteriskcentered}
\labelitemiv 1954 \newcommand\labelitemiv{\textperiodcentered}

```

■description 環境

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

```

1955 \newenvironment{description}{%
1956   \list{}{}
1957   \labelwidth=\leftmargin

```

```

1958 \labelsep=1zw
1959 \advance \labelwidth by -\labelsep
1960 \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。

```

1961 \newcommand*\descriptionlabel[1]{\normalfont\headfont #1\hfil}

```

■概要

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

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

```

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

```

```

1994         \list{}{%
1995             \listparindent\parindent
1996             \itemindent \listparindent
1997             \rightmargin \leftmargin}%
1998         \item\relax
1999     \fi
2000 \else
2001     \small
2002     \begin{center}%
2003         {\headfont \abstractname\vspace{-.5em}\vspace{\z@}}%
2004     \end{center}%
2005     \list{}{%
2006         \listparindent\parindent
2007         \itemindent \listparindent
2008         \rightmargin \leftmargin}%
2009     \item\relax
2010 \fi}{\if@twocolumn
2011     \ifx\maketitle\relax
2012     \else
2013         \endlist\end{minipage}\egroup
2014     \fi
2015 \else
2016     \endlist
2017 \fi}
2018 \fi
2019 </article | report | kiyou>
2020 <*jspf>
2021 \newbox\@abstractbox
2022 \newenvironment{abstract}{%
2023     \global\setbox\@abstractbox\hbox\bgroup
2024     \begin{minipage}[b]{157\jsc@mmm}{\sffamily Abstract}\par
2025         \small
2026         \if@english \parindent6\jsc@mmm \else \parindent1zw \fi}%
2027     {\end{minipage}\egroup}
2028 </jspf>

```

■キーワード

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

```

2029 <*jspf>
2030 %\newbox\@keywordsbox
2031 %\newenvironment{keywords}{%
2032 %     \global\setbox\@keywordsbox\hbox\bgroup
2033 %     \begin{minipage}[b]{1570\jsc@mmm}{\sffamily Keywords:}\par
2034 %         \small\parindent0zw}%
2035 %     {\end{minipage}\egroup}
2036 </jspf>

```

■verse 環境

`verse (env.)` 詩のための `verse` 環境です。

```
2037 \newenvironment{verse}{%
2038   \let \=\@centercr
2039   \list{}{%
2040     \itemsep \z@
2041     \itemindent -2zw % 元: -1.5em
2042     \listparindent\itemindent
2043     \rightmargin \z@
2044     \advance\leftmargin 2zw}% 元: 1.5em
2045   \item\relax}{\endlist}
```

■quotation 環境

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

```
2046 \newenvironment{quotation}{%
2047   \list{}{%
2048     \listparindent\parindent
2049     \itemindent\listparindent
2050     \rightmargin \z@}%
2051   \item\relax}{\endlist}
```

■quote 環境

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

```
2052 \newenvironment{quote}%
2053   {\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。

```
2054 \def\@begintheorem#1#2{\trivlist\labelsep=1zw
2055   \item[\hskip \labelsep{\headfont #1\ #2}]}
2056 \def\@opargbegintheorem#1#2#3{\trivlist\labelsep=1zw
2057   \item[\hskip \labelsep{\headfont #1\ #2 (#3)}]}
```

`titlepage (env.)` タイトルを独立のページに出力するのに使われます。

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

```

2058 \newenvironment{titlepage}{%
2059 <book> \pltx@cleartooddpage %% 2017-02-24
2060 \if@twocolumn
2061 \@restonecoltrue\onecolumn
2062 \else
2063 \@restonecolfalse\newpage
2064 \fi
2065 \thispagestyle{empty}%
2066 \ifodd\c@page\setcounter{page}\@ne\else\setcounter{page}\z@\fi %% 2017-02-24
2067 }%
2068 {\if@restonecol\twocolumn \else \newpage \fi
2069 \if@twoside\else
2070 \setcounter{page}\@ne
2071 \fi}

```

■付録

\appendix 本文と付録を分離するコマンドです。

```

2072 <!*book&!report>
2073 \newcommand{\appendix}{\par
2074 \setcounter{section}{0}%
2075 \setcounter{subsection}{0}%
2076 \gdef\presectionname{\appendixname}%
2077 \gdef\postsectionname{}}
2078 % \gdef\thesection{\@Alph\c@section}% [2003-03-02]
2079 \gdef\thesection{\presectionname\@Alph\c@section\postsectionname}%
2080 \gdef\thesubsection{\@Alph\c@section.\@arabic\c@subsection}}
2081 </!*book&!report>
2082 <*book | report>
2083 \newcommand{\appendix}{\par
2084 \setcounter{chapter}{0}%
2085 \setcounter{section}{0}%
2086 \gdef\@chapapp{\appendixname}%
2087 \gdef\@chappos{}}
2088 \gdef\thechapter{\@Alph\c@chapter}}
2089 </book | report>

```

8.4 パラメータの設定

■array と tabular 環境

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

```
2090 \setlength\arraycolsep{5\jsc@empt}
```

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

```
2091 \setlength\tabcolsep{6\jsc@empt}
```

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

```
2092 \setlength\arrayrulewidth{.4\jsc@empt}
```

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

```
2093 \setlength\doublerulesep{2\jsc@empt}
```

■`tabbing` 環境

`\tabbingsep` `\'` コマンドで入るアキです。

```
2094 \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。

```
2095 \skip\@mpfootins = \skip\footins
```

■`framebox` 環境

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

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

```
2096 \setlength\fboxsep{3\jsc@empt}
```

```
2097 \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。

```
2098 <!\book&!report>\renewcommand \theequation {\@arabic\c@equation}
```

```
2099 <*\book | report>
```

```
2100 \@addtoreset{equation}{chapter}
```

```
2101 \renewcommand\theequation
```

```
2102 {\ifnum \c@chapter>\z@ \thechapter.\fi \@arabic\c@equation}
```

```
2103 </\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。

```
2104 % \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。

```
2105 % \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。

```
2106 % \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。

```
2107 <*!book&!report>
2108 \newcounter{figure}
2109 \renewcommand \thefigure {\@arabic\c@figure}
2110 <!/book&!report>
2111 <*book | report>
2112 \newcounter{figure}[chapter]
2113 \renewcommand \thefigure
2114     {\ifnum \c@chapter>\z@ \thechapter.\fi \@arabic\c@figure}
2115 </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 2116 \def\fps@figure{tbp}
2117 \def\ftype@figure{1}
\fnum@figure 2118 \def\ext@figure{lof}
2119 \def\fnum@figure{\figurename\nobreak\thefigure}
```

`figure (env.)` * 形式は段抜きのフロートです。

```
figure* (env.) 2120 \newenvironment{figure}%
2121                 {\@float{figure}}%
2122                 {\end@float}
2123 \newenvironment{figure*}%
```

```

2124          {\@dblfloat{figure}}}%
2125          {\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。

```

2126 < *!book&!report>
2127 \newcounter{table}
2128 \renewcommand\thetable{\@arabic\c@table}
2129 < /!book&!report>
2130 < *book | report>
2131 \newcounter{table}[chapter]
2132 \renewcommand \thetable
2133     {\ifnum \c@chapter>\z@ \thechapter.\fi \@arabic\c@table}
2134 < /book | report>

```

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

```

\ext@table 2135 \def\fps@table{tbp}
2136 \def\ftype@table{2}
\fnun@table 2137 \def\ext@table{lot}
2138 \def\fnun@table{\tablename\nobreak\thetable}

```

`table (env.)` * は段抜きのフロートです。

```

table* (env.) 2139 \newenvironment{table}%
2140             {\@float{table}}%
2141             {\end@float}
2142 \newenvironment{table*}%
2143             {\@dblfloat{table}}%
2144             {\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 `\belowcaptionskip` ていましたので、キャプションを表の上につけた場合にキャプションと表がくっついてしま `\belowcaptionskip` うのを直しました。

```

2145 \newlength\abovecaptionskip
2146 \newlength\belowcaptionskip
2147 \setlength\abovecaptionskip{5\jsc@empt} % 元: 10\p@
2148 \setlength\belowcaptionskip{5\jsc@empt} % 元: 0\p@

```

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

[2003-11-05] ロジックを少し変えてみました。

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

```
2149 <!*jspf>
2150 % \long\def\@makecaption#1#2{\small
2151 %   \advance\leftskip10\jsc@mmm
2152 %   \advance\rightskip10\jsc@mmm
2153 %   \vskip\abovcaptionskip
2154 %   \sbox\@tempboxa{#1\hskip1zw\relax #2}%
2155 %   \ifdim \wd\@tempboxa >\hsize
2156 %     #1\hskip1zw\relax #2\par
2157 %   \else
2158 %     \global \@minipagefalse
2159 %     \hb@xt@\hsize{\hfil\box\@tempboxa\hfil}%
2160 %   \fi
2161 %   \vskip\belowcaptionskip}}
2162 \long\def\@makecaption#1#2{\small
2163   \advance\leftskip .0628\linewidth
2164   \advance\rightskip .0628\linewidth
2165   \vskip\abovcaptionskip
2166   \sbox\@tempboxa{#1\hskip1zw\relax #2}%
2167   \ifdim \wd\@tempboxa <\hsize \centering \fi
2168   #1{\hskip1zw\relax}#2\par
2169   \vskip\belowcaptionskip}}
2170 </!jspf>
2171 <*jspf>
2172 \long\def\@makecaption#1#2{%
2173   \vskip\abovcaptionskip
2174   \sbox\@tempboxa{\small\sffamily #1\quad #2}%
2175   \ifdim \wd\@tempboxa >\hsize
2176     {\small\sffamily
2177       \list{#1}{%
2178         \renewcommand{\makelabel}[1]{##1\hfil}
2179         \itemsep \z@
2180         \itemindent \z@
2181         \labelsep \z@
2182         \labelwidth 11\jsc@mmm
2183         \listparindent\z@
2184         \leftmargin 11\jsc@mmm}\item\relax #2\endlist}
2185   \else
2186     \global \@minipagefalse
2187     \hb@xt@\hsize{\hfil\box\@tempboxa\hfil}%
2188   \fi
2189   \vskip\belowcaptionskip}
2190 </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 2191 \DeclareOldFontCommand{\mc}{\normalfont\mcfamily}{\mathmc}
\rm 2192 \DeclareOldFontCommand{\gt}{\normalfont\gtfamily}{\mathgt}
\sf 2193 \DeclareOldFontCommand{\rm}{\normalfont\rmfamily}{\mathrm}
\sf 2194 \DeclareOldFontCommand{\sf}{\normalfont\sffamily}{\mathsf}
\tt 2195 \DeclareOldFontCommand{\tt}{\normalfont\ttfamily}{\mathtt}
```

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

```
2196 \DeclareOldFontCommand{\bf}{\normalfont\bfseries}{\mathbf}
```

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

```
2197 \DeclareOldFontCommand{\it}{\normalfont\itshape}{\mathit}
2198 \DeclareOldFontCommand{\sl}{\normalfont\slshape}{\@nomath\sl}
2199 \DeclareOldFontCommand{\sc}{\normalfont\scshape}{\@nomath\sc}
```

`\cal` 数式モード以外では何もしません（警告を出します）。

```
\mit 2200 \DeclareRobustCommand*{\cal}{\@fontswitch\relax\mathcal}
2201 \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。

```
2202 \newcommand\@pnumwidth{1.55em}
2203 \newcommand\@tocrmarg{2.55em}
2204 \newcommand\@dotsep{4.5}
2205 \ifbook&!report\setcounter{tocdepth}{2}
2206 \ifbook|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)

```
2207 \newdimen\jsc@tocl@width
2208 \newcommand{\tableofcontents}{%
2209 \ifbook|report
2210 \settowidth\jsc@tocl@width{\headfont\prechaptername\postchaptername}%
2211 \settowidth\@tempdima{\headfont\appendixname}%
2212 \ifdim\jsc@tocl@width<\@tempdima \setlength\jsc@tocl@width{\@tempdima}\fi
2213 \ifdim\jsc@tocl@width<2zw \divide\jsc@tocl@width by 2 \advance\jsc@tocl@width 1zw\fi
2214 \if@twocolumn
2215 \if@restonecoltrue\onecolumn
2216 \else
2217 \if@restonecolfalse
2218 \fi
2219 \chapter*{\contentsname}%
2220 \mkboth{\contentsname}{}%
2221 \ifbook|report}
```

```

2222 <*!book&!report>
2223 \settowidth\jsc@tocl@width{\headfont\presectionname\postsectionname}%
2224 \settowidth\@tempdima{\headfont\appendixname}%
2225 \ifdim\jsc@tocl@width<\@tempdima\relax\setlength\jsc@tocl@width{\@tempdima}\fi
2226 \ifdim\jsc@tocl@width<2zw \divide\jsc@tocl@width by 2 \advance\jsc@tocl@width 1zw\fi
2227 \section*{\contentsname}%
2228 \@mkboth{\contentsname}{\contentsname}%
2229 </!book&!report>
2230 \@starttoc{toc}%
2231 <book | report> \if@restonecol\twocolumn\fi
2232 }

```

\l@part 部の目次です。

```

2233 \newcommand*{\l@part}[2]{%
2234 \ifnum \c@tocdepth >-2\relax
2235 <!book&!report> \addpenalty\@secpenalty
2236 <book | report> \addpenalty{-\@highpenalty}%
2237 \addvspace{2.25em \@plus\jsc@empt}%
2238 \begingroup
2239 \parindent \z@
2240 % \@pnumwidth should be \@tocrmarg
2241 % \rightskip \@pnumwidth
2242 \rightskip \@tocrmarg
2243 \parfillskip -\rightskip
2244 {\leavevmode
2245 \large \headfont
2246 \setlength\@lnumwidth{4zw}%
2247 #1\hfil \hb@xt@\@pnumwidth{\hss #2}}\par
2248 \nobreak
2249 <book | report> \global\@nobreaktrue
2250 <book | report> \everypar{\global\@nobreakfalse\everypar{}}%
2251 \endgroup
2252 \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)

```

2253 <*book | report>
2254 \newcommand*{\l@chapter}[2]{%
2255 \ifnum \c@tocdepth >\m@ne
2256 \addpenalty{-\@highpenalty}%
2257 \addvspace{1.0em \@plus\jsc@empt}
2258 % \vskip 1.0em \@plus\p@ % book.cls では↑がこうなっている
2259 \begingroup
2260 \parindent\z@
2261 % \rightskip\@pnumwidth
2262 \rightskip\@tocrmarg
2263 \parfillskip-\rightskip
2264 \leavevmode\headfont

```



```

2265 % \if@english\setlength\@lnumwidth{5.5em}\else\setlength\@lnumwidth{4.683zw}\fi
2266 \setlength\@lnumwidth{\jsc@tocl@width}\advance\@lnumwidth 2.683zw
2267 \advance\leftskip\@lnumwidth \hskip-\leftskip
2268 #1\nobreak\hfil\nobreak\hbox to\@pnumwidth{\hss#2}\par
2269 \penalty\@highpenalty
2270 \endgroup
2271 \fi}
2272 </book | report>

```

\l@section 節の目次です。

```

2273 <*\book&!report>
2274 \newcommand*{\l@section}[2]{%
2275   \ifnum \c@tocdepth >\z@
2276     \addpenalty{\@secpenalty}%
2277     \addvspace{1.0em \@plus\jsc@mpt}%
2278     \begingroup
2279       \parindent\z@
2280 %     \rightskip\@pnumwidth
2281     \rightskip\@tocrmarg
2282     \parfillskip-\rightskip
2283     \leavevmode\headfont
2284     %\setlength\@lnumwidth{4zw}% 元 1.5em [2003-03-02]
2285     \setlength\@lnumwidth{\jsc@tocl@width}\advance\@lnumwidth 2zw
2286     \advance\leftskip\@lnumwidth \hskip-\leftskip
2287     #1\nobreak\hfil\nobreak\hbox to\@pnumwidth{\hss#2}\par
2288     \endgroup
2289   \fi}
2290 </!\book&!report>

```

インデントと幅はそれぞれ 1.5em, 2.3em でしたが³, 1zw, 3.683zw に変えました。

```

2291 <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 2292 <*\book&!report>
2293 % \newcommand*{\l@subsection} {\@dottedtocline{2}{1.5em}{2.3em}}
2294 % \newcommand*{\l@subsubsection}{\@dottedtocline{3}{3.8em}{3.2em}}
2295 % \newcommand*{\l@paragraph} {\@dottedtocline{4}{7.0em}{4.1em}}
2296 % \newcommand*{\l@subparagraph} {\@dottedtocline{5}{10em}{5em}}
2297 %
2298 % \newcommand*{\l@subsection} {\@dottedtocline{2}{1zw}{3zw}}
2299 % \newcommand*{\l@subsubsection}{\@dottedtocline{3}{2zw}{3zw}}
2300 % \newcommand*{\l@paragraph} {\@dottedtocline{4}{3zw}{3zw}}
2301 % \newcommand*{\l@subparagraph} {\@dottedtocline{5}{4zw}{3zw}}
2302 %
2303 \newcommand*{\l@subsection}{%

```

```

2304 \tempdima\jsc@tocl@width \advance\tempdima -1zw
2305 \dottedtocline{2}{\tempdima}{3zw}}
2306 \newcommand*{\l@subsubsection}{%
2307 \tempdima\jsc@tocl@width \advance\tempdima 0zw
2308 \dottedtocline{3}{\tempdima}{4zw}}
2309 \newcommand*{\l@paragraph}{%
2310 \tempdima\jsc@tocl@width \advance\tempdima 1zw
2311 \dottedtocline{4}{\tempdima}{5zw}}
2312 \newcommand*{\l@subparagraph}{%
2313 \tempdima\jsc@tocl@width \advance\tempdima 2zw
2314 \dottedtocline{5}{\tempdima}{6zw}}
2315 </!book&!report>
2316 <*book | report>
2317 % \newcommand*{\l@section} {\dottedtocline{2}{3.8em}{3.2em}}
2318 % \newcommand*{\l@subsubsection}{\dottedtocline{3}{7.0em}{4.1em}}
2319 % \newcommand*{\l@paragraph} {\dottedtocline{4}{10em}{5em}}
2320 % \newcommand*{\l@subparagraph} {\dottedtocline{5}{12em}{6em}}
2321 \newcommand*{\l@section}{%
2322 \tempdima\jsc@tocl@width \advance\tempdima -1zw
2323 \dottedtocline{1}{\tempdima}{3.683zw}}
2324 \newcommand*{\l@section}{%
2325 \tempdima\jsc@tocl@width \advance\tempdima 2.683zw
2326 \dottedtocline{2}{\tempdima}{3.5zw}}
2327 \newcommand*{\l@subsubsection}{%
2328 \tempdima\jsc@tocl@width \advance\tempdima 6.183zw
2329 \dottedtocline{3}{\tempdima}{4.5zw}}
2330 \newcommand*{\l@paragraph}{%
2331 \tempdima\jsc@tocl@width \advance\tempdima 10.683zw
2332 \dottedtocline{4}{\tempdima}{5.5zw}}
2333 \newcommand*{\l@subparagraph}{%
2334 \tempdima\jsc@tocl@width \advance\tempdima 16.183zw
2335 \dottedtocline{5}{\tempdima}{6.5zw}}
2336 </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。

```

2337 \newdimen\@lnumwidth
2338 \def\numberline#1{\hb@xt{\@lnumwidth}{#1\hfil}\hspace{0pt}}

```

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

[2018-06-23] デフォルトでは のようにベースラインになります。

これを変更可能にするため、`\jsTocLine` というマクロに切り出しました。例えば、仮想ボディの中央 に変更したい場合は

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

とします。

```
2339 \def\jsTocLine{\leaders\hbox{%
2340   $\m@th \mkern \@dotsep mu\hbox{.}\mkern \@dotsep mu$\hfill}
2341 \def\@dottedtocline#1#2#3#4#5{\ifnum #1>\c@tocdepth \else
2342   \vskip \z@ \@plus.2\jsc@mpt
2343   {\leftskip #2\relax \rightskip \@tocrmarg \parfillskip -\rightskip
2344     \parindent #2\relax\@afterindenttrue
2345     \interlinepenalty\@M
2346     \leavevmode
2347     \@lnumwidth #3\relax
2348     \advance\leftskip \@lnumwidth \null\nobreak\hskip -\leftskip
2349     {#4}\nobreak
2350     \jsTocLine \nobreak\hb@xt@\@pnumwidth{%
2351       \hfil\normalfont \normalcolor #5}\par}\fi}
```

■ 図目次と表目次

`\listoffigures` 図目次を出力します。

```
2352 \newcommand{\listoffigures}{%
2353   <*book | report>
2354   \if@twocolumn\@restonecoltrue\onecolumn
2355   \else\@restonecolfalse\fi
2356   \chapter*{\listfigurename}%
2357   \@mkboth{\listfigurename}{}%
2358   </book | report>
2359   <!*book&!report>
2360   \section*{\listfigurename}%
2361   \@mkboth{\listfigurename}{\listfigurename}%
2362   <!/book&!report>
2363   \@starttoc{lof}%
2364   <book | report> \if@restonecol\twocolumn\fi
2365 }
```

`\l@figure` 図目次の項目を出力します。

```
2366 \newcommand*{\l@figure}{\@dottedtocline{1}{1zw}{3.683zw}}
```

`\listoftables` 表目次を出力します。

```
2367 \newcommand{\listoftables}{%
2368   <*book | report>
2369   \if@twocolumn\@restonecoltrue\onecolumn
2370   \else\@restonecolfalse\fi
2371   \chapter*{\listtablename}%
2372   \@mkboth{\listtablename}{}%
2373   </book | report>
2374   <!*book&!report>
2375   \section*{\listtablename}%
2376   \@mkboth{\listtablename}{\listtablename}%
2377   <!/book&!report>
```

```

2378 \starttoc{lot}%
2379 <book | report> \if@restonecol\twocolumn\fi
2380 }

```

`\l@table` 表目次は図目次と同じです。

```

2381 \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。

```

2382 \newdimen\bibindent
2383 \setlength\bibindent{2zw}

```

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

```

2384 \newenvironment{thebibliography}[1]{%
2385 \global\let\presectionname\relax
2386 \global\let\postsectionname\relax
2387 <article | jsf> \section*{\refname}\@mkboth{\refname}{\refname}%
2388 <*kiyou>
2389 \vspace{1.5\baselineskip}
2390 \subsubsection*{\refname}\@mkboth{\refname}{\refname}%
2391 \vspace{0.5\baselineskip}
2392 </kiyou>
2393 <book | report> \chapter*{\bibname}\@mkboth{\bibname}{}%
2394 <book | report> \addcontentsline{toc}{chapter}{\bibname}%
2395 \list{\@biblabel{\@arabic\c@enumiv}}%
2396 {\settowidth\labelwidth{\@biblabel{#1}}%
2397 \leftmargin\labelwidth
2398 \advance\leftmargin\labelsep
2399 \@openbib@code
2400 \usecounter{enumiv}%
2401 \let\p@enumiv\@empty
2402 \renewcommand\theenumiv{\@arabic\c@enumiv}}%
2403 <kiyou> \small
2404 \sloppy
2405 \clubpenalty4000
2406 \@clubpenalty\clubpenalty
2407 \widowpenalty4000%
2408 \sfcode'\.\@m}
2409 {\def\@noitemerr
2410 {\@latex@warning{Empty 'thebibliography' environment}}}%
2411 \endlist}

```

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

```

2412 \newcommand{\newblock}{\hspace{.11em}\@plus.33em\@minus.07em}

```

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

```
2413 \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。

```
2414 % \def\@biblabel#1{\inhibitglue [#1] \inhibitglue}
```

`\cite` 文献の番号を出力する部分は ltbib1.dtx で定義されていますが、コンマとカッコを和文 `\@cite` フォントにするには次のようにします。とりあえずコメントアウトしておきましたので、必要に応じて生かしてください。かっこの前後に入るグルーを `\inhibitglue` で取っているので、オリジナル同様、Knuth[~]`\cite{knu}` のように半角空白で囲んでください。

```
2415 % \def\@citex[#1]#2{\leavevmode
2416 %   \let\@citea\empty
2417 %   \@cite{\@for\@citeb:=#2\do
2418 %     {\@citea\def\@citea{\inhibitglue\penalty\@m\ }%
2419 %     \edef\@citeb{\expandafter\@firstofone\@citeb\@empty}%
2420 %     \if@filesw\immediate\write\@auxout{\string\citation{\@citeb}}\fi
2421 %     \ifundefined{b@\@citeb}{\mbox{\normalfont\bfseries ?}%
2422 %       \G@refundefinedtrue
2423 %       \@latex@warning
2424 %         {Citation ‘\@citeb’ on page \thepage \space undefined}}%
2425 %     {\@citeofmt{\csname b@\@citeb\endcsname}}}\@citex[#1]}
2426 % \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。

```
2427 % \DeclareRobustCommand\cite{\unskip
2428 %   \@ifnextchar [{\@tempwatrue\@citex}{\@tempwafalse\@citex[]}]
2429 % \def\@cite#1#2{$\sim\hbox{\scriptsize{#1\if@tempswa
2430 %   , \inhibitglue\ #2\fi}} }$}
```

10.3 索引

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

```
2431 \newenvironment{theindex}{% 索引を 3 段組で出力する環境
2432   \if@twocolumn
2433     \onecolumn\@restonecolfalse
2434   \else
2435     \clearpage\@restonecoltrue
2436   \fi
2437   \columnseprule.4pt \columnsep 2zw
2438   \ifx\multicols\@undefined
2439 <book | report>   \twocolumn[\@makeschapterhead{\indexname}]%
2440 <book | report>   \addcontentsline{toc}{chapter}{\indexname}]%
2441 <!book&!report>   \def\presectionname{\def\postsectionname}%
2442 <!book&!report>   \twocolumn[\section*{\indexname}]%
```

```

2443 \else
2444 \ifdim\textwidth<\fullwidth
2445 \setlength{\evensidemargin}{\oddsidemargin}
2446 \setlength{\textwidth}{\fullwidth}
2447 \setlength{\linewidth}{\fullwidth}
2448 <book | report> \begin{multicols}{3}[chapter*{\indexname}]%
2449 <book | report> \addcontentsline{toc}{chapter}{\indexname}]%
2450 <!book&!report> \def\presectionname{}\def\postsectionname{}%
2451 <!book&!report> \begin{multicols}{3}[section*{\indexname}]%
2452 \else
2453 <book | report> \begin{multicols}{2}[chapter*{\indexname}]%
2454 <book | report> \addcontentsline{toc}{chapter}{\indexname}]%
2455 <!book&!report> \def\presectionname{}\def\postsectionname{}%
2456 <!book&!report> \begin{multicols}{2}[section*{\indexname}]%
2457 \fi
2458 \fi
2459 <book | report> \@mkboth{\indexname}{}%
2460 <!book&!report> \@mkboth{\indexname}{\indexname}%
2461 \plainifnotempty % \thispagestyle{plain}
2462 \parindent\z@
2463 \parskip\z@ \@plus .3\jsc@mpt\relax
2464 \let\item\@idxitem
2465 \raggedright
2466 \footnotesize\narrowbaselines
2467 }{
2468 \ifx\multicols\@undefined
2469 \if@restonecol\onecolumn\fi
2470 \else
2471 \end{multicols}
2472 \fi
2473 \clearpage
2474 }

```

`\@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` 2475 `\newcommand{\@subitem}{\par\hangindent 4zw} % 元 40pt`
`\subsubitem` 2476 `\newcommand{\@subsubitem}{\@subitem \hspace*{2zw}} % 元 20pt`
2477 `\newcommand{\@subsubitem}{\@subitem \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。

```
2478 \newcommand{\indexspace}{\par \vskip 10\jsc@mpt \@plus5\jsc@mpt \@minus3\jsc@mpt\relax}
```

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

`\alsoname` *see also* という英語ですが、ここではとりあえず両方とも「→」に変えました。⇒
 $(\$ \rightarrow \$)$ などでもいいでしょう。

```
2479 \newcommand\seename{\if@english see\else →\fi}
2480 \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。

```
2481 \@ifl@t@r\pfmtversion{2016/09/03}
2482     {\jsc@needspace@tchfalse}{\jsc@needspace@tchtrue}
2483 \ifjsc@needspace@tch
2484     \let\footnotes@ve=\footnote
2485     \def\footnote{\inhibitglue\footnotes@ve}
2486     \let\footnotemarks@ve=\footnotemark
2487     \def\footnotemark{\inhibitglue\footnotemarks@ve}
2488 \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 の変更に従いました (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。

```
2489 \@ifl@t@r\pfmtversion{2016/04/17}
2490     {\jsc@needspace@tchfalse}{\jsc@needspace@tchtrue}
2491 \ifjsc@needspace@tch
2492 \renewcommand\@makefnmark{%
2493     \ifdir \hbox{}\hbox{\@textsuperscript{\normalfont\@thefnmark}}\hbox{}%
2494     \else\hbox{\yoko\@textsuperscript{\normalfont\@thefnmark}}\fi}
2495 \fi
```

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

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

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

```
2496 \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。

```
2497 % \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。

```
2498 \renewcommand{\footnoterule}{%
2499   \kern-3\jsc@empt
2500   \hrule width .4\columnwidth height 0.4\jsc@empt
2501   \kern 2.6\jsc@empt}
```

`\c@footnote` 脚注番号は章ごとにリセットされます。

```
2502 <book | report>\@addtoreset{footnote}{chapter}
```

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

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

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

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

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

[2022-09-13] L^AT_EX 2_ε 2021-11-15 (lfloat.dtx 2021/10/14 v1.2g) で `\@currentcounter` が追加されましたので、追随します。なお、L^AT_EX 2_ε 2021-06-01 (lfloat.dtx 2021/02/10 v1.2e) で parhook 対応として `\par` が追加されていますが、実は同時に `\color@endgroup` も `\endgraf` するように変更されていますので、不要だと思います。というわけで追加しません。

[2025-04-27] `\footnote` の内容の先頭に `\inhibitglue\ignorespaces` が挿入されるようにしました。

```
2503 \long\def\@footnotetext{%
2504   \insert\footins\bgroup
2505     \normalfont\footnotesize
2506     \interlinepenalty\interfootnotelinepenalty
2507     \splittopskip\footnotesep
2508     \splitmaxdepth \dp\strutbox \floatingpenalty \@MM
2509     \hsize\columnwidth \@parboxrestore
2510     \def\@currentcounter{footnote}%
2511     \protected@edef\@currentlabel{%
2512       \csname p@footnote\endcsname\@thefnmark
2513     }%
2514     \color@begingroup
2515       \@makefntext{%
2516         \rule{z@{footnotesep}\ignorespaces}%
2517         \futurelet\jsc@next\jsc@fo@t}
2518 \def\jsc@fo@t{\ifcat\bgroup\noexpand\jsc@next \let\jsc@next\jsc@fo@t
2519               \else \let\jsc@next\jsc@f@t\fi \jsc@next}
2520 \def\jsc@f@t@t@prefix{\inhibitglue\ignorespaces}
2521 \def\jsc@f@t@t\bgroup\aftergroup\jsc@@foot\afterassignment\jsc@f@t@t@prefix\let\jsc@next}
2522 \def\jsc@f@t#1{#1\jsc@@foot}
2523 \def\jsc@@foot{\@finalstrut\strutbox\color@endgroup\egroup
2524   \ifx\pltx@foot@penalty\@undefined\else
```



```

2525 \ifhmode\null\fi
2526 \ifnum\pltx@foot@penalty=\z@\else
2527 \penalty\pltx@foot@penalty
2528 \pltx@foot@penalty\z@
2529 \fi
2530 \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。

```

2531 \newcommand\@makefntext[1]{%
2532 \advance\leftskip 3zw
2533 \parindent 1zw
2534 \noindent
2535 \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。

```

2536 % \def\@xfootnotenext[#1]{%
2537 % \begingroup
2538 % \ifnum#1>\z@
2539 % \csname c@\@mpfn\endcsname #1\relax
2540 % \unrestored@protected@xdef\@thefnmark{\thempfn}%
2541 % \else
2542 % \unrestored@protected@xdef\@thefnmark{}%
2543 % \fi
2544 % \endgroup
2545 % \@footnotetext}

```

11 段落の頭へのグルー挿入禁止

段落頭のかぎっこなどを見かけ 1 字半下げから全角 1 字下げに直します。

[2016-07-18] `\inhibitglue` の発行対象を `\inhibitxspcode` が 2 に設定されているものすべてに拡大しました。

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

[2017-02-13] `\jsc@tempa` は実はテンポラリーではなく「この処理専用のユニーク制御綴」である必要があります。間違って別の箇所です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)。

```

2546 \def\@inhibitglue{%
2547 \futurelet\@let@token\@@inhibitglue}
2548 \begingroup

```

```

2549 \let\GDEF=\gdef
2550 \let\CATCODE=\catcode
2551 \let\ENDGROUP=\endgroup
2552 \CATCODE'k=12
2553 \CATCODE'a=12
2554 \CATCODE'n=12
2555 \CATCODE'j=12
2556 \CATCODE'i=12
2557 \CATCODE'c=12
2558 \CATCODE'h=12
2559 \CATCODE'r=12
2560 \CATCODE't=12
2561 \CATCODE'e=12
2562 \GDEF\KANJI@CHARACTER{kanji character }
2563 \ENDGROUP
2564 \def\@@inhibitglue{%
2565   \expandafter\expandafter\expandafter\jsc@inhibitglue\expandafter\meaning\expandafter\@let@
2566   \expandafter\def\expandafter\jsc@inhibitglue\expandafter#\expandafter1\KANJI@CHARACTER#2#3\j
2567   \def\jsc@ig@temp{#1}%
2568   \ifx\jsc@ig@temp\@empty
2569     \ifnum\the\inhibitxspcode'#2=2\relax
2570       \inhibitglue
2571     \fi
2572   \fi}
2573 \let\everyparhook=\@inhibitglue
2574 \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。

```

2575 \def\@doendpe{%
2576   \@endpetrue
2577   \def\par{%
2578     \@restorepar\clubpenalty\@clubpenalty\everypar{\everyparhook}\par\@endpefalse}%
2579   \everypar{\setbox\z@\lastbox}\everypar{\everyparhook}\@endpefalse\everyparhook}}

```

[2017-08-31] `minipage` 環境にも対策します。

```

2580 \def\@setminipage{%
2581   \@minipagetrue
2582   \everypar{\@minipagefalse\everypar{\everyparhook}}%
2583 }

```

`\item` 命令の直後です。

```

2584 \def\@item[#1]{%
2585   \if@noperitem
2586     \@donoperitem
2587   \else

```

```

2588 \if@inlabel
2589 \indent \par
2590 \fi
2591 \ifhmode
2592 \unskip\unskip \par
2593 \fi
2594 \if@newlist
2595 \if@nobreak
2596 \@nbitem
2597 \else
2598 \addpenalty\@beginparpenalty
2599 \addvspace\@topsep
2600 \addvspace{-\parskip}%
2601 \fi
2602 \else
2603 \addpenalty\@itempenalty
2604 \addvspace\itemsep
2605 \fi
2606 \global\@inlabeltrue
2607 \fi
2608 \everypar{%
2609 \@minipagefalse
2610 \global\@newlistfalse
2611 \if@inlabel
2612 \global\@inlabelfalse
2613 {\setbox\z@\lastbox
2614 \ifvoid\z@
2615 \kern-\itemindent
2616 \fi}%
2617 \box\@labels
2618 \penalty\z@
2619 \fi
2620 \if@nobreak
2621 \@nobreakfalse
2622 \clubpenalty \@M
2623 \else
2624 \clubpenalty \@clubpenalty
2625 \everypar{\everyparhook}%
2626 \fi\everyparhook}%
2627 \if@noitemarg
2628 \@noitemargfalse
2629 \if@nmbrlist
2630 \refstepcounter\@listctr
2631 \fi
2632 \fi
2633 \sbox\@tempboxa{\makelabel{#1}}%
2634 \global\setbox\@labels\hbox{%
2635 \unhbox\@labels
2636 \hskip \itemindent

```

```

2637 \hskip -\labelwidth
2638 \hskip -\labelsep
2639 \ifdim \wd\@tempboxa >\labelwidth
2640 \box\@tempboxa
2641 \else
2642 \hbox to\labelwidth {\unhbox\@tempboxa}%
2643 \fi
2644 \hskip \labelsep}%
2645 \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。

```

2646 \def\@afterheading{%
2647 \@nobreake true
2648 \everypar{%
2649 \if@nobreake
2650 \@nobreake false
2651 \clubpenalty \@M
2652 \if@afterindent \else
2653 {\setbox\z@\lastbox}%
2654 \fi
2655 \else
2656 \clubpenalty \@clubpenalty
2657 \everypar{\everyparhook}%
2658 \fi\everyparhook}}

```

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

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

```

2659 \def\@gnewline #1{%
2660 \ifvmode
2661 \@nolnerr
2662 \else
2663 \unskip \reserved@a {\reserved@f#1}\nobreake \hfil \break \null
2664 \inhibitglue \ignorespaces
2665 \fi}

```

12 いろいろなロゴ

$\text{L}\text{A}\text{T}\text{E}\text{X}$ 関連のロゴを作り直します。

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

`\小` 文字を小さめに出したり上寄りに小さめに出したりする命令を、`jslogo.sty` では名称変更 `\上小`

してありますので、コピーします。

```
2666 \if@jslogo
2667   \IfFileExists{jslogo.sty}{%
2668     \RequirePackage{jslogo}%
2669     \def\小{\jslg@small}%
2670     \def\上小{\jslg@uppersmall}%
2671   }{%
2672     \ClassWarningNoLine{\jsc@clsname}{%
2673       The redefinitions of LaTeX-related logos has\MessageBreak
2674       been moved to jslogo.sty since 2016, but\MessageBreak
2675       jslogo.sty not found. Current release of\MessageBreak
2676       'jsclasses' includes it, so please check\MessageBreak
2677       the installation}%
2678   }
2679 \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: しっぱ愛好家さん)。

[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。

```
2680 %\let\ltx@ifnextchar\@ifnextchar
2681 %\def\ProvidesFile#1{%
2682 %  \begingroup
2683 %    \catcode'\ 10 %
2684 %    \ifnum \endlinechar<256 %
2685 %      \ifnum \endlinechar>\m@ne
2686 %        \catcode\endlinechar 10 %
2687 %      \fi
2688 %    \fi
2689 %    \@makeother\/%
2690 %    \@makeother\&%
2691 %    \ltx@ifnextchar[{\@providesfile{#1}}{\@providesfile{#1}[]}]}
```

14 初期設定

■いろいろな語

```
\prepartname
\postpartname
\prechaptername
\postchaptername
\presectionname
\postsectionname
```

```

2692 \newcommand{\prepartname}{\if@english Part~\else 第\fi}
2693 \newcommand{\postpartname}{\if@english\else 部\fi}
2694 <book | report>\newcommand{\prechaptername}{\if@english Chapter~\else 第\fi}
2695 <book | report>\newcommand{\postchaptername}{\if@english\else 章\fi}
2696 \newcommand{\presectionname}{}% 第
2697 \newcommand{\postsectionname}{}% 節

\contentsname

\listfigurename 2698 \newcommand{\contentsname}{\if@english Contents\else 目次\fi}
\listtablename 2699 \newcommand{\listfigurename}{\if@english List of Figures\else 図目次\fi}
2700 \newcommand{\listtablename}{\if@english List of Tables\else 表目次\fi}

\refname

\bibname 2701 \newcommand{\refname}{\if@english References\else 参考文献\fi}
\indexname 2702 \newcommand{\bibname}{\if@english Bibliography\else 参考文献\fi}
2703 \newcommand{\indexname}{\if@english Index\else 索引\fi}

\figurename

\tablename 2704 <!jspf>\newcommand{\figurename}{\if@english Fig.~\else 図\fi}
2705 <jspf>\newcommand{\figurename}{Fig.~}
2706 <!jspf>\newcommand{\tablename}{\if@english Table~\else 表\fi}
2707 <jspf>\newcommand{\tablename}{Table~}

\appendixname

\abstractname 2708 % \newcommand{\appendixname}{\if@english Appendix~\else 付録\fi}
2709 \newcommand{\appendixname}{\if@english\else 付録\fi}
2710 <!book>\newcommand{\abstractname}{\if@english Abstract\else 概要\fi}

```

■今日の日付 L^AT_EX で処理した日付を出力します。和暦にするには **和暦** と書いてください。ちなみにこの文章の作成日は西暦では 2025 年 4 月 27 日で、和暦では令和 7 年 4 月 27 日です。

```

\today

2711 \newif\if 西暦 \西暦 true
2712 \def\西暦{\西暦 true}
2713 \def\和暦{\西暦 false}
2714 \newcount\heisei \heisei\year \advance\heisei-1988\relax
2715 \def\pltx@today@year@#1{%
2716   \ifnum\numexpr\year-#1=1 元\else
2717     \ifnum1=\ifttdir\ifmdir0\else1\fi\else0\fi
2718     \kansuji\numexpr\year-#1\relax
2719   \else
2720     \number\numexpr\year-#1\relax\nobreak
2721   \fi
2722 \fi 年
2723 }
2724 \def\pltx@today@year{%
2725   \ifnum\numexpr\year*10000+\month*100+\day<19890108

```

```

2726 昭和 \pltx@today@year@{1925}%
2727 \else\ifnum\numexpr\year*10000+\month*100+\day<20190501
2728 平成 \pltx@today@year@{1988}%
2729 \else
2730 令和 \pltx@today@year@{2018}%
2731 \fi\fi}
2732 \def\today{%
2733 \if@english
2734 \ifcase\month\or
2735 January\or February\or March\or April\or May\or June\or
2736 July\or August\or September\or October\or November\or December\fi
2737 \space\number\day, \number\year
2738 \else\if 西暦
2739 \ifnum1=\iftdir\ifmdir0\else1\fi\else0\fi \kansuji\year
2740 \else\number\year\nobreak\fi 年
2741 \else
2742 \pltx@today@year
2743 \fi
2744 \ifnum1=\iftdir\ifmdir0\else1\fi\else0\fi
2745 \kansuji\month 月
2746 \kansuji\day 日
2747 \else
2748 \number\month\nobreak 月
2749 \number\day\nobreak 日
2750 \fi\fi}

```

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

```

2751 \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。

```

2752 <article>\if@slide \pagestyle{empty} \else \pagestyle{plain} \fi
2753 <book>\if@report \pagestyle{plain} \else \pagestyle{headings} \fi
2754 <report|kiyou>\pagestyle{plain}
2755 <jspf>\pagestyle{headings}
2756 \pagenumbering{arabic}
2757 \if@twocolumn
2758 \twocolumn
2759 \sloppy
2760 \flushbottom
2761 \else
2762 \onecolumn
2763 \raggedbottom
2764 \fi
2765 \if@slide
2766 \renewcommand\kanjifamilydefault{\gtdefault}
2767 \renewcommand?familydefault{\sfdefault}
2768 \raggedright

```

```
2769 \xkanjiskip=0.1em\relax
2770 \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。

```
2771 < *article | book | report >
2772 \ifpackageloaded{exppl2e}{\jsc@needspace@tchtrue}{\jsc@needspace@tchfalse}
2773 \ifjsc@needspace@tch\else
2774 \expandafter\endinput
2775 \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
2776 \def\@gnewline #1{%
2777 \ifvmode
2778 \@nolnerr
2779 \else
2780 \unskip \reserved@a {\reserved@f#1}\nobreak \hfil \break \hspace \z@
2781 \ignorespaces
2782 \fi}
2783 < /article | book | report >
2784 < /class >
```

以上です。