

Lua \LaTeX -ja 用 jsclasses 互換クラス

Lua \TeX -ja プロジェクト

2025-04-27

目次

1	はじめに	2
1.1	jsclasses.dtx からの主な変更点	2
2	Lua \TeX -ja の読み込み	4
3	オプション	4
4	和文フォントの変更	15
5	フォントサイズ	18
6	レイアウト	23
6.1	ページレイアウト	24
7	改ページ (日本語 \TeX 開発コミュニティ版のみ)	31
8	ページスタイル	32
9	文書のマークアップ	35
9.1	表題	35
9.2	章・節	39
9.3	リスト環境	51
9.4	パラメータの設定	57
9.5	フロート	58
9.6	キャプション	60
10	フォントコマンド	61
11	相互参照	63
11.1	目次の類	63
11.2	参考文献	68

11.3	索引	70
11.4	脚注	71
12	段落の頭へのグルー挿入禁止	73
13	いろいろなロゴ	75
14	初期設定	79

1 はじめに

これは、元々奥村晴彦先生により作成され、現在は日本語 \TeX 開発コミュニティにより管理されている `jsclasses.dtx` を `Lua \TeX -ja` 用に改変したものです。次のドキュメントクラス（スタイルファイル）を生成します。

[2017-02-13] `forum:2121` の議論を機に、`ltjsreport` クラスを新設しました。従来の `ltjsbook` 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>ltjsarticle.cls</code>	論文・レポート用
<code><book></code>	<code>ltjsbook.cls</code>	書籍用
<code><report></code>	<code>ltjsreport.cls</code>	レポート用
<code><jspf></code>	<code>ltjspf.cls</code>	某学会誌用
<code><kiyou></code>	<code>ltjskiyou.cls</code>	某紀要用

1.1 `jsclasses.dtx` からの主な変更点

全ての変更点を知りたい場合は、`jsclasses.dtx` と `ltjsclasses.dtx` で `diff` をとって下さい。 `zw`, `zh` は全て `\zw`, `\zh` に置き換えられています。

- フォントメトリック関係のオプション `winjis` は単に無視されます。
- 標準では `jfm-ujis.lua` (`Lua \TeX -ja` 標準のメトリック、`OTF` パッケージのものがベース) を使用します。
- `uplatex` オプション、`autodetect-engine` オプションを削除してあります（前者ではエラーを出すようにしています）。
- `disablejfam` オプションはクラス側では何もしません（ただ `Lua \TeX -ja` 本体に渡されるだけです）。もし

```
! LaTeX Error: Too many math alphabets used in version ****.
```

のエラーが起こった場合は、`lualatex-math` パッケージを読み込んでみて下さい。

- `papersize` オプションの指定に関わらず `PDF` のページサイズは適切に設定されます。
- `Lua \TeX -ja` 同梱のメトリックを用いる限りは、段落の頭にグルーは挿入されません。そのため、オリジナルの `jsclasses` 内にあった `hack` (`\everyparhook`) は不要になったので、削除しました。

- 「amsmath との衝突の回避」のコードは、上流で既に対処されているうえ、これがあると `grfext.sty` を読み込んだ際にエラーを引き起こすので削除しました。
- 本家 `jsclasses` では `\mag` を用いて「10 pt 時の組版結果を本文フォントサイズに合わせ拡大縮小」という方針でしたが、本 `ltjsclasses` ではそのような方法を取っていません。
 - `nomag` オプション指定時には、単にレイアウトに用いる各種長さの値をスケールさせるだけです。そのため、例えば本文の文字サイズが 17 pt のときには `cmr10` でなく `cmr17` を用いることになり、組版結果の印象が異なる恐れがあります。
 - `nomag*` オプション指定時には、上記に加えてオプティカルサイズを調整する（本文では `cmr17` の代わりに `cmr10` を拡大縮小する、など）ため、 \LaTeX のフォント選択システム NFSS ヘパッチを当てます。こうすることで前項に書いた不具合はなくなりますが、かえって別の不具合が起きる可能性はあります^{*1}。
 標準では `nomag*` オプションが有効になっています。`jsclasses` で用意され、かつ既定になっている `usemag` オプションを指定すると警告を出します。

[2014-02-07 LTJ] `jsclasses` 2014-02-07 ベースにしました。

[2014-07-26 LTJ] 縦組用和文フォントの設定を加えました。

[2014-12-24 LTJ] `\@setfontsize` 中の和欧文間空白の設定で `if` 文が抜けていたのを直しました。

[2016-01-30 LTJ] `\rmfamily` 他で和文フォントファミリーも変更するコードを \LaTeX -ja カーネル内に移しました。

[2016-03-21 LTJ] \LaTeX beta-0.87.0 では PDF 出力時に `\mag` が使用できなくなったので、ZR さんの `bxjscls` を参考に使わないように書き換えました。

[2016-03-31 LTJ] `xreal` オプションを標準で有効にしました。

[2016-07-12 LTJ] `jsclasses` 開発版に合わせ、`real`、`xreal` オプションの名称を変更するなどの変更を行いました。

[2016-07-18 LTJ] `usemag` オプションが指定されると警告を出すようにしました。

[2016-07-21 LTJ] \LaTeX 等のロゴの再定義で、`jslogo` パッケージがあればそちらを読み込むことにしました。

[2016-10-13 LTJ] `slide` オプションの使用時にエラーが出るのを修正。

以下では実際のコードに即して説明します。

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

```
1 %<article>\def\jsc@clsname{ltjsarticle}
2 %<book>\def\jsc@clsname{ltjsbook}
3 %<report>\def\jsc@clsname{ltjsreport}
4 %<jspf>\def\jsc@clsname{ltjspf}
5 %<kiyou>\def\jsc@clsname{ltjskiyou}
```

^{*1} `nomag*` は `jsclasses` でも利用可能ですが、`ltjsclasses` では `jsclasses` とは別の実装をしています。

2 LuaTeX-ja の読み込み

和文スケール値を設定した後に、LuaTeX-ja を読み込みます。

```
6 %<!jspf>\def\Cjascale{0.924715}
7 %<jspf>\def\Cjascale{0.903375}
8 \RequirePackage{luatexja}
```

3 オプション

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

```
9 \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。

```
10 \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。

```
11 %<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。

```
12 %<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。

```
13 %<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が、実際には用いられません。

```
14 \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, 182mm×230mm), a4var (A4 変形, 210mm×283mm) を追加しました。

```
15 \DeclareOption{a3paper}{%
16   \setlength\paperheight {420mm}%
17   \setlength\paperwidth  {297mm}}
18 \DeclareOption{a4paper}{%
19   \setlength\paperheight {297mm}%
20   \setlength\paperwidth  {210mm}}
21 \DeclareOption{a5paper}{%
22   \setlength\paperheight {210mm}%
23   \setlength\paperwidth  {148mm}}
24 \DeclareOption{a6paper}{%
25   \setlength\paperheight {148mm}%
26   \setlength\paperwidth  {105mm}}
27 \DeclareOption{b4paper}{%
28   \setlength\paperheight {364mm}%
29   \setlength\paperwidth  {257mm}}
30 \DeclareOption{b5paper}{%
31   \setlength\paperheight {257mm}%
32   \setlength\paperwidth  {182mm}}
33 \DeclareOption{b6paper}{%
34   \setlength\paperheight {182mm}%
35   \setlength\paperwidth  {128mm}}
36 \DeclareOption{a4j}{%
37   \setlength\paperheight {297mm}%
38   \setlength\paperwidth  {210mm}}
39 \DeclareOption{a5j}{%
40   \setlength\paperheight {210mm}%
41   \setlength\paperwidth  {148mm}}
42 \DeclareOption{b4j}{%
43   \setlength\paperheight {364mm}%
44   \setlength\paperwidth  {257mm}}
45 \DeclareOption{b5j}{%
46   \setlength\paperheight {257mm}%
47   \setlength\paperwidth  {182mm}}
48 \DeclareOption{a4var}{%
49   \setlength\paperheight {283mm}%
50   \setlength\paperwidth  {210mm}}
51 \DeclareOption{b5var}{%
52   \setlength\paperheight {230mm}%
53   \setlength\paperwidth  {182mm}}
54 \DeclareOption{letterpaper}{%
55   \setlength\paperheight {11in}%
56   \setlength\paperwidth  {8.5in}}
57 \DeclareOption{legalpaper}{%
58   \setlength\paperheight {14in}%
59   \setlength\paperwidth  {8.5in}}
60 \DeclareOption{executivepaper}{%
```

```
61 \setlength\paperheight {10.5in}%
62 \setlength\paperwidth {7.25in}}
```

■横置き 用紙の縦と横の長さを入れ換えます。

```
63 \newif\if@landscape
64 \@landscapefalse
65 \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
ため article のみで使えるオプションとしました。

```
66 \newif\if@slide
67 \@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をご迷惑をおかけしましたが, 標準的なドキュメントクラスと同様にポイント数から 10 を引いたものに直しました。

[2003-03-22] 14Q オプションを追加しました。

[2003-04-18] 12Q オプションを追加しました。

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

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

```
68 \def\jsc@magscale{1}
69 %<*article>
70 \DeclareOption{slide}{\@slidetrue\def\jsc@magscale{3.583}\@landscapetrue\@titlepagetrue}
71 %</article>
72 \DeclareOption{8pt} {\def\jsc@magscale{0.833}}% 1.2-1
73 \DeclareOption{9pt} {\def\jsc@magscale{0.913}}% 1.2-0.5
74 \DeclareOption{10pt}{\def\jsc@magscale{1}}
75 \DeclareOption{11pt}{\def\jsc@magscale{1.095}}% 1.20.5
76 \DeclareOption{12pt}{\def\jsc@magscale{1.200}}
77 \DeclareOption{14pt}{\def\jsc@magscale{1.440}}
78 \DeclareOption{17pt}{\def\jsc@magscale{1.728}}
79 \DeclareOption{20pt}{\def\jsc@magscale{2}}
80 \DeclareOption{21pt}{\def\jsc@magscale{2.074}}
81 \DeclareOption{25pt}{\def\jsc@magscale{2.488}}
82 \DeclareOption{30pt}{\def\jsc@magscale{2.986}}
83 \DeclareOption{36pt}{\def\jsc@magscale{3.583}}
84 \DeclareOption{43pt}{\def\jsc@magscale{4.300}}
85 \DeclareOption{12Q} {\def\jsc@magscale{0.923}}% 1pt*12Q/13Q
86 \DeclareOption{14Q} {\def\jsc@magscale{1.077}}% 1pt*14Q/13Q
87 \DeclareOption{10ptj}{\def\jsc@magscale{1.085}}% 1pt*10bp/13Q
```

```

88 \DeclareOption{10.5ptj}{\def\jsc@magscale{1.139}}
89 \DeclareOption{11ptj}{\def\jsc@magscale{1.194}}
90 \DeclareOption{12ptj}{\def\jsc@magscale{1.302}}

```

■**オブティカルサイズの補正** `nomag*` オプション指定時には、本文のフォントサイズが 10pt 以外の場合にオブティカルサイズの補正を行うために NFSS にパッチを当てます。現在の `ltjclasses` ではこのパッチ当ては標準では行いますが、将来どうなるかわからないので `nomag` で無効化することができるようにしました。

`noxreal`, `real` は旧来の互換性として今は残してありますが、2017 年 7 月に削除する予定です。

[2018-01-14] `noxreal`, `real` を削除しました。また、内部命令の名称を `jsclasses` に合わせました。

```

91 \newif\ifjsc@mag@xreal
92 \jsc@mag@xrealtrue
93 \DeclareOption{nomag*}{\jsc@mag@xrealtrue}
94 \DeclareOption{nomag}{\jsc@mag@xrealfalse}
95 \DeclareOption{usemag}{%
96   \ClassWarningNoLine{\jsc@clsname}{%
97     This \jsc@clsname\space cls does not support `usemag'\MessageBreak
98     option, since LuaTeX does not support \string\mag\MessageBreak in pdf output}%
99   \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 `lltjcore.sty`で行います。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 `luatexja-compat.sty` で宣言されています。

```

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

```

114 \DeclareOption{mentuke}{%
115   \tombowtrue \tombowdatefalse

```

```

116 \setlength{\tombowwidth}{\z@}%
117 \maketombowbox}

```

■**両面，片面オプション** `twoside` で奇数ページ・偶数ページのレイアウトが変わります。

[2003-04-29] `vartwoside` でどちらのページも傍注が右側になります。

```

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

```

■**二段組** `twocolumn` で二段組になります。

```

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

```

123 \DeclareOption{titlepage}{\@titlepagetrue}
124 \DeclareOption{notitlepage}{\@titlepagefalse}

```

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

[2017-02-24] `openright` は横組では奇数ページ起こし，縦組では偶数ページ起こしを表します。ややこしいですが，これは L^AT_EX 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， `latexclasses` では新たに `openleft` も追加しました。

```

125 %<book|report>\DeclareOption{openright}{\@openrighttrue\@openleftfalse}
126 %<book|report>\DeclareOption{openleft}{\@openlefttrue\@openrightfalse}
127 %<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。

```

128 \def\eqnarray{%
129   \stepcounter{equation}%
130   \def\@currentlabel{\p@equation\theequation}%
131   \global\@eqnswtrue
132   \m@th
133   \global\@eqcnt\z@
134   \tabskip\@centering
135   \let\\\@eqncr
136   $$\everycr{\halign to\displaywidth\bgroup
137     \hskip\@centering$\displaystyle\tabskip\z@skip{##}$\@eqnsel
138     &\global\@eqcnt\@ne \hfil$\displaystyle{##}$\hfil
139     &\global\@eqcnt\tw@ $\displaystyle{##}$\hfil\tabskip\@centering
140     &\global\@eqcnt\thr@@ \hb@xt@\z@\bgroup\hss##\egroup

```



```

141     \tabskip\z@skip
142     \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。

```

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

```

■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。

```

170 % \DeclareOption{openbib}{%
171 %     \AtEndOfPackage{%
172 %         \renewcommand\@openbib@code{%
173 %             \advance\leftmargin\bibindent
174 %             \itemindent -\bibindent
175 %             \listparindent \itemindent
176 %             \parsep \z@}%
177 %         \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 pTeX では数式中では 16 通り
のフォントしか使えませんでした。 LuaTeX では Omega 拡張が取り込まれていて 256 通
りのフォントが使えます。ただし、L^AT_EX 2_ε カーネルでは未だに数式ファミリの数は 16 個に

制限されているので、実際に使用可能な数式ファミリの数を増やすためには `lualatex-math` パッケージを読み込む必要があることに注意が必要です。

[2018-10-08 LTJ] LuaTeX-ja 本体が `disablejfam` オプションをサポートしたので、クラスファイルからは削除します。

[2019-08-12 LTJ] ……と思いましたが、“Unused global option(s): [disablejfam]” 警告が出てしまいますので、「何もしない」クラスオプションとして形だけ定義します。

```
178 %<*article|report|book>
179 \DeclareOption{disablejfam}{}
180 %</article|report|book>
```

■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。

```
181 \DeclareOption{draft}{\setlength\overfullrule{5pt}}
182 \DeclareOption{final}{\setlength\overfullrule{0pt}}
```

■和文フォントメトリックの選択 ここでは OTF パッケージのメトリックを元とした、`jfm-ujis.lua` メトリックを標準で使います。古い `min10`, `goth10` 互換のメトリックを使いたいときは `mingoth` というオプションを指定します。`pTeX` でよく利用される `jis` フォントメトリックと互換のメトリックを使いたい場合は、`ptexjis` というオプションを指定します。`winjis` メトリックは用済みのため、`winjis` オプションは無視されます。

[2016-11-09] `pLaTeX` / `upLaTeX` を自動判別するオプション `autodetect-engine` を新設しました。

[2016-11-24 LTJ] `autodetect-engine` は LuaTeX-ja では意味がないので警告を表示させます。

[2018-07-30 LTJ] `uplatex` 指定時のエラーが正しく表示されなかったので修正しました。

```
183 \newif\ifmingoth
184 \mingothfalse
185 \newif\ifjisfont
186 \jisfontfalse
187 \newif\ifptexjis
188 \ptexjisfalse
189 \DeclareOption{winjis}{%
190   \ClassWarningNoLine{\jsc@clsname}{this class does not support `winjis' option}}
191 \DeclareOption{uplatex}{%
192   \ClassError{\jsc@clsname}{this class does not support `uplatex' option}}
193 \DeclareOption{autodetect-engine}{%
194   \ClassWarningNoLine{\jsc@clsname}{this class does not support `autodetect-
195     engine' option}}
195 \DeclareOption{mingoth}{\mingothtrue}
196 \DeclareOption{ptexjis}{\ptexjisttrue}
197 \DeclareOption{jis}{\jisfonttrue}
```

■papersize スペシャルの利用 `ltjclasses` では `papersize` オプションの有無に関わらず、PDF のページサイズは適切に設定されるので、削除しました。

■英語化 オプション `english` を新設しました。

```
198 \newif\if@english
199 \@englishfalse
200 \DeclareOption{english}{\@englishtrue}
```

■`ltjsbook` を `ltjsreport` もどきに オプション `report` を新設しました。

[2017-02-13] 従来は「`ltjsreport` 相当」を `ltjsbook` の `report` オプションで提供していましたが、新しく `ltjsreport` 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。

```
201 %<*book>
202 \newif\if@report
203 \@reportfalse
204 \DeclareOption{report}{\@reporttrue\@openrightfalse\@twosidefalse\@mparswitchfalse}
205 %</book>
```

■`jslogo` パッケージの読み込み `LATEX` 関連のロゴを再定義する `jslogo` パッケージを読み込まないオプション `nojslogo` を新設しました。`jslogo` オプションの指定で従来どおりの動作となります。デフォルトは `jslogo` で、すなわちパッケージを読み込みます。

```
206 \newif\if@jslogo \@jslogotrue
207 \DeclareOption{jslogo}{\@jslogotrue}
208 \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。

```
209 %<article>\ExecuteOptions{a4paper,oneside,onecolumn,notitlepage,final}
210 %<book>\ExecuteOptions{a4paper,twoside,onecolumn,titlepage,openright,final}
211 %<report>\ExecuteOptions{a4paper,oneside,onecolumn,titlepage,openany,final}
212 %<jspf>\ExecuteOptions{a4paper,twoside,twocolumn,notitlepage,fleqn,final}
213 %<kiyou>\ExecuteOptions{a4paper,twoside,twocolumn,notitlepage,final}
214 \ProcessOptions
```

後処理

```
215 \if@slide
216 \def\maybeblue{\@ifundefined{ver@color.sty}{\color{blue}}{}}
217 \fi
218 \if@landscape
219 \setlength\@tempdima {\paperheight}
220 \setlength\paperheight{\paperwidth}
221 \setlength\paperwidth {\@tempdima}
222 \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。

```
223 %<article|book|report>\if@slide\def\n@baseline{13}\else\def\n@baseline{16}\fi
224 %<jspf>\def\n@baseline{14.554375}
225 %<kiyou>\def\n@baseline{14.897}
```

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

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

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

[2016-03-21 LTJ] `\mag` を使わないように全面的に書き換えました。`\ltjs@empt` に「拡大率だけ大きくした pt」の値が格納されます。bxjcls と同様に、`\@ptsize` は 10pt, 11pt, 12pt オプションが指定された時だけ従来通り 0, 1, 2 と設定し、それ以外の場合は -20 とすることにしました。`\inv@mag` はもはや定義していません。

[2016-03-26 LTJ] `\ltjs@magscale` に拡大率を格納した後、それをを用いて `\ltjs@empt` を設定するようにしました。

[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。

[2016-07-12 LTJ] `\ltjs@...` を本家に合わせて `\jsc@...` に名称変更しました。

```
226 %<*kiyou>
227 \def\jsc@magscale{0.9769230}
228 %</kiyou>
229 \newdimen\jsc@empt
230 \newdimen\jsc@mmm
231 \jsc@empt=\jsc@magscale\p@
232 \jsc@mmm=\jsc@magscale mm
233 \ifdim\jsc@empt<.92\p@ % 8pt, 9pt 指定時
234 \def\n@baseline{15}%
235 \fi
236 \newcommand{\@ptsize}{0}
237 \ifdim\jsc@empt=1.0954\p@ \renewcommand{\@ptsize}{1}\else
238 \ifdim\jsc@empt=1.2\p@ \renewcommand{\@ptsize}{2}\else
239 \renewcommand{\@ptsize}{-20}\fi\fi
```

■**オプティカルサイズの補正**

[2016-03-26 LTJ] `xreal` オプションの指定時には、bxjcls の `magstyle=xreal` オプションのように、オプティカルの補正を行うために NFSS にパッチを当てます。パッチは、概ね misc さんによる「js*.cls 同様の文字サイズ設定を `\mag` によらずに行う方法：試案」(<http://oku.edu.mie-u.ac.jp/~okumura/texfaq/qa/28416.html>) の方法に沿っていますが、拡大/縮小するところの計算には Lua を用いています。

なお、T_EX 内部で長さは sp 単位の整数倍で表現されているので、数 sp の誤差は仕方ないです。そのため、事前に `type1cm` パッケージを読みこんでおきます。

[2016-03-28 LTJ] `\luafunction` を使うようにし、また本文のフォントサイズが 10pt のときには（不要なので）パッチを当てないことにしました。

[2016-04-04 LTJ] NFSS へのパッチを修正。

[2017-01-23 LTJ] L^AT_EX 2_ε 2017-01-01 以降では TU エンコーディングが標準なので、type1cm パッケージは読み込まないようにしました。

[2017-02-17 LTJ] \directlua 中で出力される数字のカテゴリコードが 12 になるようにしました。この保証をしないと例えば listings パッケージで無限ループになります。

[2018-07-02 LTJ] 10pt オプションが指定されており、実際にはオプティカルサイズの補正が不要なときは「xreal オプションは指定されなかった」という扱いにしておきます。

```
240 \ifjsc@mag@xreal
241 \ifdim\jsc@mpt=\p@\jsc@mag@xrealfalse\else
242 \expandafter\let\csname OT1/cmr/m/n/10\endcsname\relax
243 \expandafter\let\csname TU/lmr/m/n/10\endcsname\relax
244 \expandafter\let\csname OMX/cmex/m/n/10\endcsname\relax
245 \newluafunction\ltjs@@magnify@font@calc
246 \begingroup\catcode`\%=12\relax
247 \directlua{
248   local getdimen, mpt=tex.getdimen, tex.getdimen('jsc@mpt')/65536
249   local floor, cnt_dimen_at = math.floor, luatexbase.registernumber 'dimen@'
250   local print_scaled, sprint = luatexja.print_scaled, tex.sprint
251   local t = lua.get_functions_table()
252   t[\the\ltjs@@magnify@font@calc] = function()
253     sprint(-2, floor(0.5+mpt*getdimen(cnt_dimen_at)))
254   end
255   function luatexja.ltjs_unmagnify_fsize(a)
256     local s = print_scaled(floor(0.5+a/mpt*65536))
257     sprint(-2, (s:match('%.$')) and s:sub(1,-3) or s )
258   end
259 }
260 \endgroup
261 \def\ltjs@magnify@external@font#1 at#2 at#3\@nil{%
262   \def\@tempa{#1}\def\@tempb{#2}%
263   \ifx\@tempb\@empty
264     \edef\@tempb{ scaled\directlua{%
265       tex.sprint(-2,math.floor(0.5+\jsc@magscale*1000))
266     }}%
267   \else
268     \dimen@\@tempb\relax
269     \edef\@tempb{ at\luafunction\ltjs@@magnify@font@calc sp}%
270   \fi
271   \edef\@tempa{\def\noexpand\external@font{\@tempa\@tempb}}%
272 }
273 \let\ltjs@orig@get@external@font=\get@external@font
274 \def\get@external@font{%
275   \edef\@f@size{\directlua{luatexja.ltjs_unmagnify_fsize(\@f@size)}}%
276   \ltjs@orig@get@external@font
277 \begingroup
278   \edef\@tempa{\external@font\space at\space at}%
279   \expandafter\ltjs@magnify@external@font\@tempa\@nil
```

```

280 \expandafter\endgroup\@tempa
281 }
282 \fi\fi

```

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

```

\jsc@smallskip
\jsc@medskip 283 \def\jsc@smallskip{\vspace\jsc@smallskipamount}
\jsc@bigskip 284 %\def\jsc@medskip{\vspace\jsc@medskipamount}
285 %\def\jsc@bigskip{\vspace\jsc@bigskipamount}

```

```

\jsc@smallskipamount
\jsc@medskipamount 286 \newskip\jsc@smallskipamount
\jsc@bigskipamount 287 \jsc@smallskipamount=3\jsc@empt plus 1\jsc@empt minus 1\jsc@empt
288 %\newskip\jsc@medskipamount
289 \jsc@medskipamount =6\jsc@empt plus 2\jsc@empt minus 2\jsc@empt
290 %\newskip\jsc@bigskipamount
291 %\jsc@bigskipamoun =12\jsc@empt plus 4\jsc@empt minus 4\jsc@empt

```

■PDF の用紙サイズの設定

`\pagewidth` 出力の PDF の用紙サイズをここで設定しておきます。tombow が真のときは 2 インチ足し
`\pageheight` しておきます。

`\stockwidth` [2015-10-18 LTJ] Lua_TE_X 0.81.0 ではプリミティブの名称変更がされたので、それに合
`\stockheight` わせておきます。

[2016-07-12 LTJ] luatex.def が新しくなったことに対応する aminophen さんのパッチを
取り込みました。

[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。

[2022-09-12 LTJ] L^AT_EX 2_ε カーネルに `\stockwidth`、`\stockheight` が追加されるよう
ですので、クラスファイル側では未定義のときのみこれらの長さ変数を定義します。h20y6m
さん、ありがとうございます。

```

292 \iftombow
293 \ifdefined\stockwidth\else\newlength{\stockwidth}\fi
294 \ifdefined\stockheight\else\newlength{\stockheight}\fi
295 \setlength{\stockwidth}{\paperwidth}
296 \setlength{\stockheight}{\paperheight}
297 \advance \stockwidth 2in
298 \advance \stockheight 2in
299 \ifdefined\pdfpagewidth
300 \setlength{\pdfpagewidth}{\stockwidth}
301 \setlength{\pdfpageheight}{\stockheight}
302 \else
303 \setlength{\pagewidth}{\stockwidth}

```

```

304   \setlength{\pageheight}{\stockheight}
305   \fi
306 \else
307   \ifdefined\pdfpagewidth
308     \setlength{\pdfpagewidth}{\paperwidth}
309     \setlength{\pdfpageheight}{\paperheight}
310   \else
311     \setlength{\pagewidth}{\paperwidth}
312     \setlength{\pageheight}{\paperheight}
313   \fi
314 \fi

```

4 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。

$\text{pT}_{\text{E}}\text{X}$ (アスキーが日本語化した $\text{T}_{\text{E}}\text{X}$) では, 例えば従来のフォントメトリック `min10` や JIS フォントメトリックでは「公称 10 ポイントの和文フォントは, 実際には 9.62216pt で出力される (メトリック側で 0.962216 倍される)」という仕様になっています。一方, $\text{LuaT}_{\text{E}}\text{X-j}\text{a}$ の提供するメトリックでは, そのようなことはありません。公称 10 ポイントの和文フォントは, 10 ポイントで出力されます。

この `ltjclasses` でも, 派生元の `jsclasses` と同じように, 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, $13\text{Q}/10\text{pt} \approx 0.924715$ 倍すればいいことになります。

`\ltj@stdmcfont`, `\ltj@stdgdfont` による, デフォルトで使われる明朝・ゴシックのフォントの設定に対応しました。この 2 つの命令の値はユーザが日々の利用でその都度指定するものではありません。

[2014-07-26 LTJ] なお, 現状のところ, 縦組用 JFM は `jfm-ujisv.lua` しか準備していません。

[2016-03-21 LTJ] 拡大率の計算で 1 pt を 1/72.27 インチでなく 0.3514mm と間違えて扱っていたのを修正。

[2017-12-31] 和文スケール ($1\text{zw} \div \text{要求サイズ}$) を表す実数値マクロ `\Cjascale` を定義しました。

[2020-05-30] `\ltj@stdmcfont` の説明に古い記述があったので修正。

これにより, 公称 10 ポイントの和文フォントを 0.924715 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。

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, $0.9 * 72.27/72 \approx$

0.903375 倍します。

[2018-09-23 LTJ] 実際の `\Cjascale` の定義は `LuaTeX-ja` の読み込み前に移動しました。こうすることによって「0.962216 倍された和文フォント」という実際には使われない和文フォントを読み込む必要がなくなります。

実際にフォントの再定義を行う部分です。

[2018-09-23 LTJ] `\Cjascale` の設定を前倒ししたことに伴い、実際の再定義は `mingoth`, `ptexjis` のときしか必要なくなりました。

[2022-04-11 LTJ] kern feature を無効にするのを忘れていました。

```
315 \expandafter\let\csname JY3/mc/m/n/10\endcsname\relax
316 \ifmingoth
317 \DeclareFontShape{JY3}{mc}{m}{n}{<-> s * [\Cjascale] \ltj@stdmcfont:-
    kern;jfm=min}{}
318 \DeclareFontShape{JY3}{gt}{m}{n}{<-> s * [\Cjascale] \ltj@stdgtfont:-
    kern;jfm=min}{}
319 \else
320 \ifptexjis
321 \DeclareFontShape{JY3}{mc}{m}{n}{<-> s * [\Cjascale] \ltj@stdmcfont:-
    kern;jfm=jis}{}
322 \DeclareFontShape{JY3}{gt}{m}{n}{<-> s * [\Cjascale] \ltj@stdgtfont:-
    kern;jfm=jis}{}
323 \fi
324 \fi
```

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。

[2014-03-25 LTJ] タイプライタ体に合わせるファミリーを `\jttdefault` とし、通常のゴシック体と別にできるようにしました。`\jttdefault` は、標準で `\gtdefault` と定義しています。

[2003-03-16] イタリック体、斜体について、和文でゴシックを当てていましたが、数学の定理環境などで多量のイタリック体を使うことがあり、ゴシックにすると黒々となってしまふという弊害がありました。`amsthm` を使わない場合は定理の本文が明朝になるように `\newtheorem` 環境を手直ししてしのいでいましたが、`TeX` が数学で多用されることを考えると、イタリック体に明朝体を当てたほうがい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。

[2016-01-30 LTJ] `\rmfamily`, `\sffamily`, `\ttfamily` の再定義を `LuaTeX-ja` カーネルに移動させたので、ここでは和文対応にするフラグ `\@ltj@match@family` を有効にさせるだけでよいです。

[2018-06-09 LTJ] シリーズ `b` は同じ書体の `bx` と等価になるように宣言します。

```
325 \DeclareFontShape{JY3}{mc}{bx}{n}{<->ssub*gt/m/n}{}
326 \DeclareFontShape{JY3}{gt}{bx}{n}{<->ssub*gt/m/n}{}
327 \DeclareFontShape{JY3}{mc}{b}{n}{<->ssub*gt/m/n}{}
328 \DeclareFontShape{JY3}{mc}{b}{n}{<->ssub*gt/m/n}{}
329 \DeclareFontShape{JY3}{mc}{m}{it}{<->ssub*mc/m/n}{}

```



```

330 \DeclareFontShape{JY3}{mc}{m}{sl}{<->ssub*mc/m/n}{}
331 \DeclareFontShape{JY3}{mc}{m}{sc}{<->ssub*mc/m/n}{}
332 \DeclareFontShape{JY3}{gt}{m}{it}{<->ssub*gt/m/n}{}
333 \DeclareFontShape{JY3}{gt}{m}{sl}{<->ssub*gt/m/n}{}
334 \DeclareFontShape{JY3}{mc}{bx}{it}{<->ssub*gt/m/n}{}
335 \DeclareFontShape{JY3}{mc}{bx}{sl}{<->ssub*gt/m/n}{}
336 \DeclareFontShape{JY3}{mc}{b}{it}{<->ssub*gt/m/n}{}
337 \DeclareFontShape{JY3}{mc}{b}{sl}{<->ssub*gt/m/n}{}
338 \DeclareFontShape{JT3}{mc}{bx}{n}{<->ssub*gt/m/n}{}
339 \DeclareFontShape{JT3}{gt}{bx}{n}{<->ssub*gt/m/n}{}
340 \DeclareFontShape{JT3}{mc}{b}{n}{<->ssub*gt/m/n}{}
341 \DeclareFontShape{JT3}{mc}{b}{n}{<->ssub*gt/m/n}{}
342 \DeclareFontShape{JT3}{mc}{m}{it}{<->ssub*mc/m/n}{}
343 \DeclareFontShape{JT3}{mc}{m}{sl}{<->ssub*mc/m/n}{}
344 \DeclareFontShape{JT3}{mc}{m}{sc}{<->ssub*mc/m/n}{}
345 \DeclareFontShape{JT3}{gt}{m}{it}{<->ssub*gt/m/n}{}
346 \DeclareFontShape{JT3}{gt}{m}{sl}{<->ssub*gt/m/n}{}
347 \DeclareFontShape{JT3}{mc}{bx}{it}{<->ssub*gt/m/n}{}
348 \DeclareFontShape{JT3}{mc}{bx}{sl}{<->ssub*gt/m/n}{}
349 \DeclareFontShape{JT3}{mc}{b}{it}{<->ssub*gt/m/n}{}
350 \DeclareFontShape{JT3}{mc}{b}{sl}{<->ssub*gt/m/n}{}
351 \renewcommand\jttdefault{\gtdefault}\@ltj@match@familytrue

```

LuaTeX-jā では和文組版に伴うグルーはノードベースで挿入するようになり、また欧文・和文間のグルーとイタリック補正は干渉しないようになりました。まだ「和文の斜体」については LuaLaTeX カーネル側でまともな対応がされていませんが、jclasses で行われていた `\textmc`、`\textgt` の再定義は不要のように思われます。

jclasses.dtx 中で行われていた `\reDeclareMathAlphabet` の再定義は削除。また、Yue ZHANG さん作の `fixjfm` パッケージ対応のコードも LuaTeX-jā では削除しています。

```

352 \AtBeginDocument{%
353   \unless\ifltj@disablejfam
354     \reDeclareMathAlphabet{\mathrm}{\mathrm}{\mathmc}
355     \reDeclareMathAlphabet{\mathbf}{\mathbf}{\mathgt}
356   \fi
357 }%

```

`\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。

```

358 % \DeclareTextCommand{\textsterling}{OT1}{\itshape\char`\$}

```

アスキーの `kinsoku.dtx` では「」 「“”」前後のペナルティが 5000 になっていたのを、jclasses.dtx ではそれを 10000 に補正していました。しかし、LuaTeX-jā では最初から

これらのパラメータは 10000 なので、もはや補正する必要はありません。

「TeX!」「〒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。

```
359 \ltjsetparameter{jaxspmode={`!,2}}
360 \ltjsetparameter{jaxspmode={`〒,1}}
```

「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。

```
361 \ltjsetparameter{alxspmode={`+,3}}
362 \ltjsetparameter{alxspmode={`%,3}}
```

jsclasses.dtx では 80~ff の文字の \xspcode を全て 3 にしていましたが、LuaTeX-ja では同様の内容が最初から設定されていますので、対応する部分は削除。

\@ 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。

```
363 \def\@{\spacefactor3000{}}
```

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

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

`kanjiskip` は `ltj-latex.sty` で `0pt plus 0.4pt minus 0.5pt` 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 (1\zw) に直します。

[2008-02-18] `english` オプションで `\parindent` を `1em` にしました。

[2014-05-14 LTJ] `\ltjsetparameter` の実行は時間がかかるので、`\ltjsetkanjiskip` と `\ltjsetxkanjiskip` (両者とも、実行前には `\ltj@setpar@global` の実行が必要) にしました。

[2014-12-24 LTJ] `jsclasses` では、`\@setfontsize` 中で `xkanjiskip` を設定するのは現在の和欧文間空白の自然長が正の場合だけでした。`ltjsclasses` では最初からこの判定が抜けてしまっていたので、復活させます。

```

364 \def\@setfontsize#1#2#3{%
365 % \@nomath#1%
366 \ifx\protect\@typeset@protect
367   \let\@currsz#1%
368 \fi
369 \fontsize{#2}{#3}\selectfont
370 \ifdim\parindent>\z@
371   \if@english
372     \parindent=1em
373   \else
374     \parindent=1\zw
375   \fi
376 \fi
377 \ltj@setpar@global
378 \ltjsetkanjiskip\z@ plus .1\zw minus .01\zw
379 \@tempskipa=\ltjgetparameter{xkanjiskip}
380 \ifdim\@tempskipa>\z@
381   \if@slide
382     \ltjsetxkanjiskip .1em
383   \else
384     \ltjsetxkanjiskip .25em plus .15em minus .06em
385   \fi
386 \fi}

```

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

```

387 \def\jsc@setfontsize#1#2#3{%
388   \@setfontsize#1{#2\jsc@mpt}{#3\jsc@mpt}}

```

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

```

389 \emergencystretch 3\zw

```

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

`\narrowbaselines`

`\widebaselines`

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

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

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

[2015-01-07 LTJ] 遅くなりましたが、<http://oku.edu.mie-u.ac.jp/tex/mod/forum/discuss.php?d=1005> にあった ZR さんのパッチを取り込みました。

```
390 \newif\ifnarrowbaselines
391 \if@english
392   \narrowbaselinestrue
393 \fi
394 \def\narrowbaselines{%
395   \narrowbaselinestrue
396   \skip0=\abovedisplayskip
397   \skip2=\abovedisplayshortskip
398   \skip4=\belowdisplayskip
399   \skip6=\belowdisplayshortskip
400   \@currsize\selectfont
401   \abovedisplayskip=\skip0
402   \abovedisplayshortskip=\skip2
403   \belowdisplayskip=\skip4
404   \belowdisplayshortskip=\skip6\relax}
405 \def\widebaselines{\narrowbaselinesfalse\@currsize\selectfont}
406 \def\ltj@ifnarrowbaselines{%
407   \ifnarrowbaselines\expandafter\@firstoftwo
408   \else \expandafter\@secondoftwo
409   \fi
410 }
```

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

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

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

```
411 \renewcommand{\normalsize}{%
412   \ltj@ifnarrowbaselines
413     {\jsc@setfontsize\normalsize\@xpt\@xiipt}%
414     {\jsc@setfontsize\normalsize\@xpt{\n@baseline}}%
```

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

[2003-02-16] ちょっと変えました。

[2009-08-26] T_EX Q & A 52569 から始まる議論について逡巡していましたが、結局、微調

節してみることにしました。

```
415 \abovedisplayskip 11\jsc@mpt \@plus3\jsc@mpt \@minus4\jsc@mpt
416 \abovedisplayshortskip \z@ \@plus3\jsc@mpt
417 \belowdisplayskip 9\jsc@mpt \@plus3\jsc@mpt \@minus4\jsc@mpt
418 \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。

```
419 \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。

```
420 \mcfamily\selectfont\normalsize
```

`\Cht` 基準となる長さの設定をします。11tjfont.sty で宣言されているパラメータに実際の値を `\Cdp` 設定します。たとえば `\Cwd` は `\normalfont` の全角幅 (`1\zw`) です。

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

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

```
\Chs
421 \setbox0\hbox{漢}
422 \setlength\Cht{\ht0}
423 \setlength\Cdp{\dp0}
424 \setlength\Cwd{\wd0}
425 \setlength\Cvs{\baselineskip}
426 \setlength\Chs{\wd0}
427 \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。

```
428 \newcommand{\small}{%
429 \ltj@ifnarrowbaselines
430 %<!kiyou> {\jsc@setfontsize\small\@ixpt{11}}%
431 %<kiyou> {\jsc@setfontsize\small{8.8888}{11}}%
432 %<!kiyou> {\jsc@setfontsize\small\@ixpt{13}}%
433 %<kiyou> {\jsc@setfontsize\small{8.8888}{13.2418}}%
434 \abovedisplayskip 9\jsc@mpt \@plus3\jsc@mpt \@minus4\jsc@mpt
435 \abovedisplayshortskip \z@ \@plus3\jsc@mpt
436 \belowdisplayskip \abovedisplayskip
437 \belowdisplayshortskip \belowdisplayskip
438 \def\@listi{\leftmargin\leftmargini
439 \topsep \z@
440 \parsep \z@
441 \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。

```

442 \newcommand{\footnotesize}{%
443 \ltj@ifnarrowbaselines
444 %<!kiyou> {\jsc@setfontsize\footnotesize\@viipt{9.5}}%
445 %<kiyou> {\jsc@setfontsize\footnotesize{8.8888}{11}}%
446 %<!kiyou> {\jsc@setfontsize\footnotesize\@viipt{11}}%
447 %<kiyou> {\jsc@setfontsize\footnotesize{8.8888}{13.2418}}%
448 \abovedisplayskip 6\jsc@empt \@plus2\jsc@empt \@minus3\jsc@empt
449 \abovedisplayshortskip \z@ \@plus2\jsc@empt
450 \belowdisplayskip \abovedisplayskip
451 \belowdisplayshortskip \belowdisplayskip
452 \def\@listi{\leftmargin\leftmargini
453 \topsep \z@
454 \parsep \z@
455 \itemsep \parsep}}

```

`\scriptsize` それ以外のサイズは、本文に使うことがないので、単にフォントサイズと行送りだけ変更し
`\tiny` ます。特に注意すべきは `\large` で、これは二段組のときに節見出しのフォントとして使い、
`\large` 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
`\Large` 行が揃うようにします。

`\LARGE` [2004-11-03] `\HUGE` を追加。

```

456 \newcommand{\scriptsize}{\jsc@setfontsize\scriptsize\@viipt\@viipt}
457 \newcommand{\tiny}{\jsc@setfontsize\tiny\@vpt\@vpt}
458 \if@twocolumn
459 %<!kiyou> \newcommand{\large}{\jsc@setfontsize\large\@xipt{\n@baseline}}
460 %<kiyou> \newcommand{\large}{\jsc@setfontsize\large{11.111}{\n@baseline}}
461 \else
462 %<!kiyou> \newcommand{\large}{\jsc@setfontsize\large\@xipt{17}}
463 %<kiyou> \newcommand{\large}{\jsc@setfontsize\large{11.111}{17}}
464 \fi
465 %<!kiyou>\newcommand{\Large}{\jsc@setfontsize\Large\@xivpt{21}}
466 %<kiyou>\newcommand{\Large}{\jsc@setfontsize\Large{12.222}{21}}
467 \newcommand{\LARGE}{\jsc@setfontsize\LARGE\@xviipt{25}}
468 \newcommand{\huge}{\jsc@setfontsize\huge\@xxpt{28}}
469 \newcommand{\Huge}{\jsc@setfontsize\Huge\@xxvpt{33}}
470 \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。

```

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

```

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

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。『pL^AT_EX 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。

```
472 % \newcommand{\headfont}{\bfseries}
473 \newcommand{\headfont}{\gtfamily\sffamily}
474 % \newcommand{\headfont}{\sffamily\fontseries{sbc}\selectfont}
```

6 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、`2\zw` にしました。このスペースの中央に `\columnseprule` の幅の罫線が引かれます。

```
475 %<!kiyou>\setlength\columnsep{2\zw}
476 %<kiyou>\setlength\columnsep{28truebp}
477 \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 478 \setlength\lineskip{1\jsc@empt}
\normallineskip 479 \setlength\normallineskip{1\jsc@empt}
\normallineskiplimit 480 \setlength\lineskiplimit{1\jsc@empt}
481 \setlength\normallineskiplimit{1\jsc@empt}
```

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

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

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

```
482 \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。

```
483 \setlength\parskip{\z@}
484 \if@slide
485 \setlength\parindent{0\zw}
486 \else
487 \setlength\parindent{1\zw}
488 \fi
```

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

`\@highpenalty`

```
489 \@lowpenalty 51
490 \@medpenalty 151
491 \@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。

```
492 % \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。

```
493 % \brokenpenalty 100
```

6.1 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 f のような高い文字が入ったときに 1 行目のベースラインが他のページより下がってしまいます。ここでは本文の公称フォントサイズ (10pt) にします。

[2003-06-26] `\headheight` はヘッダの高さで、元は 12pt でしたが、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。

[2016-08-17 LTJ] 1.38zw の代わりに 1.38\zh にしています。

```
494 \setlength\topskip{1.38\zh}%% from 10\jsc@mpt (2016-08-17)
495 \if@slide
496 \setlength\headheight{0\jsc@mpt}
497 \else
498 \setlength\headheight{20\jsc@mpt}%% from 2\topskip (2016-08-17); from \topskip (2003-
06-26)
499 \fi
```

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

```
500 %<*article|kiyou>
501 \if@slide
502 \setlength\footskip{\z@}
503 \else
504 \setlength\footskip{0.03367\paperheight}
505 \ifdim\footskip<\baselineskip
506 \setlength\footskip{\baselineskip}
507 \fi
508 \fi
```



```

509 %</article|kiyou>
510 %<jspf>\setlength\footskip{9\jsc@mmm}
511 %<*book>
512 \if@report
513 \setlength\footskip{0.03367\paperheight}
514 \ifdim\footskip<\baselineskip
515 \setlength\footskip{\baselineskip}
516 \fi
517 \else
518 \setlength\footskip{\z@}
519 \fi
520 %</book>
521 %<*report>
522 \setlength\footskip{0.03367\paperheight}
523 \ifdim\footskip<\baselineskip
524 \setlength\footskip{\baselineskip}
525 \fi
526 %</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)。

```

527 %<*article>
528 \if@slide
529 \setlength\headsep{0\jsc@mp}
530 \addtolength\headsep{-\topskip}%% added (2016-10-08)
531 \addtolength\headsep{10\jsc@mp}%% added (2016-10-08)
532 \else
533 \setlength\headsep{\footskip}
534 \addtolength\headsep{-\topskip}
535 \fi
536 %</article>
537 %<*book>
538 \if@report
539 \setlength\headsep{\footskip}
540 \addtolength\headsep{-\topskip}
541 \else
542 \setlength\headsep{6\jsc@mmm}
543 \addtolength\headsep{-\topskip}%% added (2016-10-08)
544 \addtolength\headsep{10\jsc@mp}%% added (2016-10-08)
545 \fi
546 %</book>
547 %<*report>
548 \setlength\headsep{\footskip}
549 \addtolength\headsep{-\topskip}
550 %</report>

```

```

551 %<*jspf>
552 \setlength\headsep{9\jsc@mmm}
553 \addtolength\headsep{-\topskip}
554 %</jspf>
555 %<*kiyou>
556 \setlength\headheight{0\jsc@mpt}
557 \setlength\headsep{0\jsc@mpt}
558 \addtolength\headsep{-\topskip}%% added (2016-10-08)
559 \addtolength\headsep{10\jsc@mpt}%% added (2016-10-08)
560 %</kiyou>

```

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

```

561 \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。

```

562 \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 `50\zw` (25 文字 × 2 段) + 段間 8mm とします。

```

563 %<*article>
564 \if@slide
565 \setlength\fullwidth{0.9\paperwidth}
566 \else
567 \setlength\fullwidth{0.76\paperwidth}
568 \fi
569 \if@twocolumn \@tempdima=2\zw \else \@tempdima=1\zw \fi
570 \divide\fullwidth\@tempdima \multiply\fullwidth\@tempdima
571 \setlength\textwidth{\fullwidth}
572 %</article>
573 %<*book>
574 \if@report
575 \setlength\fullwidth{0.76\paperwidth}
576 \else

```

```

577 \setlength\fullwidth{\paperwidth}
578 \addtolength\fullwidth{-36\jsc@mmm}
579 \fi
580 \if@twocolumn \@tempdima=2\zw \else \@tempdima=1\zw \fi
581 \divide\fullwidth\@tempdima \multiply\fullwidth\@tempdima
582 \setlength\textwidth{\fullwidth}
583 \if@report \else
584 \if@twocolumn \else
585 \ifdim \fullwidth>40\zw
586 \setlength\textwidth{40\zw}
587 \fi
588 \fi
589 \fi
590 %</book>
591 %<*report>
592 \setlength\fullwidth{0.76\paperwidth}
593 \if@twocolumn \@tempdima=2\zw \else \@tempdima=1\zw \fi
594 \divide\fullwidth\@tempdima \multiply\fullwidth\@tempdima
595 \setlength\textwidth{\fullwidth}
596 %</report>
597 %<*jspf>
598 \setlength\fullwidth{50\zw}
599 \addtolength\fullwidth{8\jsc@mmm}
600 \setlength\textwidth{\fullwidth}
601 %</jspf>
602 %<*kiyou>
603 \setlength\fullwidth{48\zw}
604 \addtolength\fullwidth{\columnsep}
605 \setlength\textwidth{\fullwidth}
606 %</kiyou>

```

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

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

某学会誌スタイルでは 44 行にします。

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

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

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

```
607 %<*article|book|report>
```

```

608 \if@slide
609 \setlength{\textheight}{0.95\paperheight}
610 \else
611 \setlength{\textheight}{0.83\paperheight}
612 \fi
613 \addtolength{\textheight}{-10\jsc@empt}%% from -\topskip (2016-10-08); from -
    \headheight (2003-06-26)
614 \addtolength{\textheight}{-\headsep}
615 \addtolength{\textheight}{-\footskip}
616 \addtolength{\textheight}{-\topskip}
617 \divide\textheight\baselineskip
618 \multiply\textheight\baselineskip
619 %</article|book|report>
620 %<jspf>\setlength{\textheight}{51\baselineskip}
621 %<kiyou>\setlength{\textheight}{47\baselineskip}
622 \addtolength{\textheight}{\topskip}
623 \addtolength{\textheight}{0.1\jsc@empt}
624 %<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 \LaTeX 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。

```

625 \def\flushbottom{%
626 \def\@textbottom{\vskip \z@ \@plus.1\jsc@empt}%
627 \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。

```

628 \setlength\marginparsep{\columnsep}
629 \setlength\marginparpush{\baselineskip}

```

`\oddsidemargin` それぞれ奇数ページ、偶数ページの左マージンから 1 インチ引いた値です。片面印刷では `\evensidemargin` `\oddsidemargin` が使われます。 \TeX は上・左マージンに `1truein` を挿入しますが、トンボ関係のオプションが指定されると `lltjcore.sty` 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。

[2011-10-03 LTJ] \LuaTeX (\pdfTeX ?) では `1truein` ではなく `1in` になるようです。

```

630 \setlength{\oddsidemargin}{\paperwidth}
631 \addtolength{\oddsidemargin}{-\fullwidth}
632 \setlength{\oddsidemargin}{.5\oddsidemargin}
633 \addtolength{\oddsidemargin}{-1in}
634 \setlength{\evensidemargin}{\oddsidemargin}
635 \if@mparswitch
636 \addtolength{\evensidemargin}{\fullwidth}

```

```
637 \addtolength{\evensidemargin}{-\textwidth}
638 \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 `1\zw` の整数倍に切り捨てます。

```
639 \setlength\marginparwidth{\paperwidth}
640 \addtolength\marginparwidth{-\oddsidemargin}
641 \addtolength\marginparwidth{-1in}
642 \addtolength\marginparwidth{-\textwidth}
643 \addtolength\marginparwidth{-10\jsc@mmm}
644 \addtolength\marginparwidth{-\marginparsep}
645 \@tempdima=1\zw
646 \divide\marginparwidth\@tempdima
647 \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。

[2011-10-03 LTJ] こども `\oddsidemargin` のときと同様に `-\inv@margin` ではなく `-1in` にします。

[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)。

```
648 \setlength\topmargin{\paperheight}
649 \addtolength\topmargin{-\textheight}
650 \if@slide
651 \addtolength\topmargin{-\headheight}
652 \else
653 \addtolength\topmargin{-10\jsc@empt}% from -\topskip (2016-10-08); from -
\headheight (2003-06-26)
654 \fi
655 \addtolength\topmargin{-\headsep}
656 \addtolength\topmargin{-\footskip}
657 \setlength\topmargin{0.5\topmargin}
658 %<kiyou>\setlength\topmargin{81truebp}
659 \addtolength\topmargin{-1in}
```

■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。

```
660 {\footnotesize}\global\setlength\footnotesep{\baselineskip}
661 \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。

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

■**フロート関連** フロート（図、表）関連のパラメータは L^AT_EX 2_ε 本体で定義されていますが、ここで設定変更します。本文ページ（本文とフロートが共存するページ）とフロートだけのページで設定が異なります。ちなみに、カウンタは内部では \c@ を名前に冠したマクロになっています。

`\c@topnumber` `topnumber` カウンタは本文ページ上部のフロートの最大数です。

[2003-08-23] ちょっと増やしました。

```
663 \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。

```
664 \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。

```
665 \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。

```
666 \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。

```
667 \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。

```
668 \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。

```
669 \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。

```
670 \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。

```
671 \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。

```
672 \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。`\intertextsep` は本文の途中に出力されるフロートと本
`\intertextsep` 文との距離です。

```
673 \setlength\floatsep {12\jsc@empt \@plus 2\jsc@empt \@minus 2\jsc@empt}
674 \setlength\textfloatsep{20\jsc@empt \@plus 2\jsc@empt \@minus 4\jsc@empt}
675 \setlength\intertextsep {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 676 \setlength\dblfloatsep {12\jsc@empt \@plus 2\jsc@empt \@minus 2\jsc@empt}
677 \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 678 \setlength\@fptop{0\jsc@empt \@plus 1fil}
679 \setlength\@fpsep{8\jsc@empt \@plus 2fil}
680 \setlength\@fpbot{0\jsc@empt \@plus 1fil}
```

`\@dblfpptop` 段抜きフロートについての値です。

```
\@dblfpsep 681 \setlength\@dblfpptop{0\jsc@empt \@plus 1fil}
682 \setlength\@dblfpsep{8\jsc@empt \@plus 2fil}
\@dblfpbot 683 \setlength\@dblfpbot{0\jsc@empt \@plus 1fil}
```

7 改ページ (日本語 TeX 開発コミュニティ版のみ)

`\pltx@cleartorightpage` [2017-02-24] コミュニティ版 pLaTeX 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。

```
684 %<*article|book|report>
685 \def\pltx@cleartorightpage{\clearpage\if@twoside
686 \unless\ifodd\numexpr\c@page+\ltjgetparameter{direction}\relax
687 \hbox{}\thispagestyle{empty}\newpage
688 \if@twocolumn\hbox{}\newpage\fi
689 \fi\fi}
690 \def\pltx@cleartoleftpage{\clearpage\if@twoside
691 \ifodd\numexpr\c@page+\ltjgetparameter{direction}\relax
692 \hbox{}\thispagestyle{empty}\newpage
693 \if@twocolumn\hbox{}\newpage\fi
694 \fi\fi}
695 \def\pltx@cleartooddpage{\clearpage\if@twoside
696 \ifodd\c@page\else
```

```

697 \hbox{\thispagestyle{empty}}\newpage
698 \if@twocolumn\hbox{\newpage}\fi
699 \fi\fi}
700 \def\pltx@cleartoevenpage{\clearpage\if@twoside
701 \ifodd\c@page
702 \hbox{\thispagestyle{empty}}\newpage
703 \if@twocolumn\hbox{\newpage}\fi
704 \fi\fi}
705 %</article|book|report>

```

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

```

706 %<*book|report>
707 \if@openleft
708 \let\cleardoublepage\pltx@cleartoleftpage
709 \else\if@openright
710 \let\cleardoublepage\pltx@cleartorightpage
711 \fi\fi
712 %</book|report>

```

8 ページスタイル

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

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

ページスタイルは `\ps@...` の形のマクロで定義されています。

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

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

```

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

```

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

```
713 % \def\ps@empty{%
714 %   \let\@mkboth\@gobbletwo
715 %   \let\@oddhead\@empty
716 %   \let\@oddfoot\@empty
717 %   \let\@evenhead\@empty
718 %   \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。

```
719 \def\ps@plainfoot{%
720   \let\@mkboth\@gobbletwo
721   \let\@oddhead\@empty
722   \def\@oddfoot{\normalfont\hfil\thepage\hfil}%
723   \let\@evenhead\@empty
724   \let\@evenfoot\@oddfoot}
725 \def\ps@plainhead{%
726   \let\@mkboth\@gobbletwo
727   \let\@oddfoot\@empty
728   \let\@evenfoot\@empty
729   \def\@evenhead{%
730     \if@mparswitch \hss \fi
731     \hbox to \fullwidth{\textbf{\thepage}\hfil}%
732     \if@mparswitch\else \hss \fi}%
733   \def\@oddhead{%
734     \hbox to \fullwidth{\hfil\textbf{\thepage}}\hss}}
735 %<book>\if@report \let\ps@plain\ps@plainfoot \else \let\ps@plain\ps@plainhead \fi
736 %<!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。

```
737 %<*article|kiyou>
738 \if@twoside
739   \def\ps@headings{%
740     \let\@oddfoot\@empty
741     \let\@evenfoot\@empty
742     \def\@evenhead{\if@mparswitch \hss \fi
743       \underline{\hbox to \fullwidth{\textbf{\thepage}\hfil\leftmark}}}%
744     \if@mparswitch\else \hss \fi}%
745     \def\@oddhead{%
746       \underline{%
747         \hbox to \fullwidth{\rightmark}\hfil\textbf{\thepage}}\hss}%
748     \let\@mkboth\markboth
```

```

749 \def\sectionmark##1{\markboth{%
750 \ifnum \c@secnumdepth >\z@ \thesection \hskip1\zw\fi
751 ##1}{}}%
752 \def\subsectionmark##1{\markright{%
753 \ifnum \c@secnumdepth >\@ne \thesubsection \hskip1\zw\fi
754 ##1}}%
755 }
756 \else % if not twoside
757 \def\ps@headings{%
758 \let\@oddfont\@empty
759 \def\@oddhead{%
760 \underline{%
761 \hbox to \fullwidth{\rightmark}\hfil\textbf{\thepage}}\hss}%
762 \let\@mkboth\markboth
763 \def\sectionmark##1{\markright{%
764 \ifnum \c@secnumdepth >\z@ \thesection \hskip1\zw\fi
765 ##1}}}
766 \fi
767 %</article|kiyou>

```

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

```

768 %<*book|report>
769 \newif\if@omit@number
770 \def\ps@headings{%
771 \let\@oddfont\@empty
772 \let\@evenfont\@empty
773 \def\@evenhead{%
774 \if@mparswitch \hss \fi
775 \underline{\hbox to \fullwidth{\ltjsetparameter{autoxspacing={true}}
776 \textbf{\thepage}\hfil\leftmark}}}%
777 \if@mparswitch\else \hss \fi}%
778 \def\@oddhead{\underline{\hbox to \fullwidth{\ltjsetparameter{autoxspacing={true}}
779 { \if@twoside\rightmark\else\leftmark\fi}\hfil\textbf{\thepage}}}\hss}%
780 \let\@mkboth\markboth
781 \def\chaptermark##1{\markboth{%
782 \ifnum \c@secnumdepth >\m@ne
783 %<book> \if@mainmatter
784 \if@omit@number\else
785 \@chapapp\thechapter\@chappos\hskip1\zw
786 \fi
787 %<book> \fi
788 \fi
789 ##1}{}}%
790 \def\sectionmark##1{\markright{%
791 \ifnum \c@secnumdepth >\z@ \thesection \hskip1\zw\fi
792 ##1}}}
793 %</book|report>

```

最後は学会誌の場合です。

```
794 %<*jspf>
795 \def\ps@headings{%
796   \def\@oddfont{\normalfont\hfil\thepage\hfil}
797   \def\@evenfont{\normalfont\hfil\thepage\hfil}
798   \def\@oddhead{\normalfont\hfil \@title \hfil}
799   \def\@evenhead{\normalfont\hfil プラズマ・核融合学会誌 \hfil}}
800 %</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。

```
801 \def\ps@myheadings{%
802   \let\@oddfont\@empty\let\@evenfont\@empty
803   \def\@evenhead{%
804     \if@mparswitch \hss \fi%
805     \hbox to \fullwidth{\thepage\hfil\leftmark}%
806     \if@mparswitch\else \hss \fi}%
807   \def\@oddhead{%
808     \hbox to \fullwidth{\rightmark\hfil\thepage}\hss}%
809   \let\@mkboth\@gobbletwo
810 %<book|report> \let\chaptermark\@gobble
811 \let\sectionmark\@gobble
812 %<!book&!report> \let\subsectionmark\@gobble
813 }
```

9 文書のマークアップ

9.1 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 814 % \newcommand*\title{1}{\gdef\@title{#1}}
815 % \newcommand*\author{1}{\gdef\@author{#1}}
816 % \newcommand*\date{1}{\gdef\@date{#1}}
817 % \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 818 %<*jspf>
\keywords 819 \newcommand*\etitle{1}{\gdef\@etitle{#1}}
820 \newcommand*\eauthor{1}{\gdef\@eauthor{#1}}
821 \newcommand*\keywords{1}{\gdef\@keywords{#1}}
822 \newcommand*\email{1}{\gdef\authors@mail{#1}}
823 \newcommand*\AuthorsEmail{1}{\gdef\authors@mail{author's e-mail:\ #1}}
824 %</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。

```

825 \def\plainifnotempty{%
826   \ifx \@oddhead \@empty
827     \ifx \@oddfont \@empty
828     \else
829       \thispagestyle{plainfoot}%
830     \fi
831   \else
832     \thispagestyle{plainhead}%
833   \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] スペーシングを元の `jsclasses` 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くなり、レイアウトが変わってしまいます。

```

834 %<*article|book|report|kiyou>
835 \if@titlepage
836   \newcommand{\maketitle}{%
837     \begin{titlepage}%
838       \let\footnotesize\small
839       \let\footnoterule\relax
840       \let\footnote\thanks
841       \null\vfil
842       \if@slide
843         {\footnotesize \@date}%
844       \begin{center}
845         \mbox{} \\\[1\zw]
846         \large
847         {\maybeblue\hrule height0\jsc@mpt depth2\jsc@mpt\relax}\par
848         \jsc@smallskip
849         \@title
850         \jsc@smallskip
851         {\maybeblue\hrule height0\jsc@mpt depth2\jsc@mpt\relax}\par
852         \vfill
853         {\small \@author}%
854       \end{center}
855     \else
856     \vskip 60\jsc@mpt
857     \begin{center}%
858       {\LARGE \@title \par}%
859     \vskip 3em%
860     {\large
861       \lineskip .75em

```

```

862     \begin{tabular}[t]{c}%
863     \@author
864     \end{tabular}\par}%
865     \vskip 1.5em
866     {\large \@date \par}%
867     \end{center}%
868     \fi
869     \par
870     \@thanks\vfil\null
871 \end{titlepage}%
872 \setcounter{footnote}{0}%
873 \global\let\thanks\relax
874 \global\let\maketitle\relax
875 \global\let\@thanks\@empty
876 \global\let\@author\@empty
877 \global\let\@date\@empty
878 \global\let\@title\@empty
879 \global\let\title\relax
880 \global\let\author\relax
881 \global\let\date\relax
882 \global\let\and\relax
883 }%
884 \else
885 \newcommand{\maketitle}{\par
886 \begingroup
887 \renewcommand\thefootnote{\@fnsymbol\c@footnote}%
888 \def\@makefnmark{\rlap{\@textsuperscript{\normalfont\@thefnmark}}}%
889 \long\def\@makefntext##1{\advance\leftskip 3\zw
890 \parindent 1\zw\noindent
891 \llap{\@textsuperscript{\normalfont\@thefnmark}\hskip0.3\zw}##1}%
892 \if@twocolumn
893 \ifnum \col@number=\@ne
894 \@maketitle
895 \else
896 \twocolumn[\@maketitle]%
897 \fi
898 \else
899 \newpage
900 \global\@topnum\z@ % Prevents figures from going at top of page.
901 \@maketitle
902 \fi
903 \plainifnotempty
904 \@thanks
905 \endgroup
906 \setcounter{footnote}{0}%
907 \global\let\thanks\relax
908 \global\let\maketitle\relax
909 \global\let\@thanks\@empty
910 \global\let\@author\@empty

```

```

911 \global\let\@date\@empty
912 \global\let\@title\@empty
913 \global\let\title\relax
914 \global\let\author\relax
915 \global\let\date\relax
916 \global\let\and\relax
917 }

```

`\@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。

```

918 \def\@maketitle{%
919 \newpage\null
920 \vskip 2em
921 \begin{center}%
922 \let\footnote\thanks
923 {\LARGE \@title \par}%
924 \vskip 1.5em
925 {\large
926 \lineskip .5em
927 \begin{tabular}[t]{c}%
928 \@author
929 \end{tabular}\par}%
930 \vskip 1em
931 {\large \@date}%
932 \end{center}%
933 \par\vskip 1.5em
934 %<article|report|kiyou> \ifvoid\@abstractbox\else\centerline{\box\@abstractbox}\vskip1.5em
935 }
936 \fi
937 %</article|book|report|kiyou>
938 %<*jspf>
939 \newcommand{\maketitle}{\par
940 \begingroup
941 \renewcommand\thefootnote{\@fnsymbol\c@footnote}%
942 \def\@makefnmark{\rlap{\@textsuperscript{\normalfont\@thefnmark}}}%
943 \long\def\@makefntext##1{\advance\leftskip 3\zw
944 \parindent 1\zw\noindent
945 \llap{\@textsuperscript{\normalfont\@thefnmark}\hskip0.3\zw}##1}%
946 \twocolumn[\@maketitle]%
947 \plainifnotempty
948 \@thanks
949 \endgroup
950 \setcounter{footnote}{0}%
951 \global\let\thanks\relax
952 \global\let\maketitle\relax
953 \global\let\@thanks\@empty
954 \global\let\@author\@empty
955 \global\let\@date\@empty
956 % \global\let\@title\@empty % \@title は柱に使う
957 \global\let\title\relax

```

```

958 \global\let\author\relax
959 \global\let\date\relax
960 \global\let\and\relax
961 \ifx\authors@mail\@undefined\else{%
962   \def\@makefnctext{\advance\leftskip 3\zw \parindent -3\zw}%
963   \footnotetext[0]{\itshape\authors@mail}%
964 } \fi
965 \global\let\authors@mail\@undefined}
966 \def\@maketitle{%
967   \newpage\null
968   \vskip 6em % used to be 2em
969   \begin{center}
970     \let\footnote\thanks
971     \ifx\@title\@undefined\else{\LARGE\headfont\@title\par} \fi
972     \lineskip .5em
973     \ifx\@author\@undefined\else
974       \vskip 1em
975       \begin{tabular}[t]{c}%
976         \@author
977       \end{tabular}\par
978     \fi
979     \ifx\@etitle\@undefined\else
980       \vskip 1em
981       {\large \@etitle \par}%
982     \fi
983     \ifx\@eauthor\@undefined\else
984       \vskip 1em
985       \begin{tabular}[t]{c}%
986         \@eauthor
987       \end{tabular}\par
988     \fi
989     \vskip 1em
990     \@date
991   \end{center}
992   \vskip 1.5em
993   \centerline{\box\@abstractbox}
994   \ifx\@keywords\@undefined\else
995     \vskip 1.5em
996     \centerline{\parbox{157\jsc@mmm}{\textsf{Keywords:}}\ \small\@keywords}
997   \fi
998   \vskip 1.5em}
999 %</jspf>

```

9.2 章・節

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

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

それぞれの引数の意味は次の通りです。

- 名** ユーザレベルコマンドの名前です (例: section)。
- レベル** 見出しの深さを示す数値です (chapter=1, section=2, ...)。この数値が `secnumdepth` 以下のとき見出し番号を出力します。
- 字下げ** 見出しの字下げ量です。
- 前アキ** この値の絶対値が見出し上側の空きです。負の場合は、見出し直後の段落をインデントしません。
- 後アキ** 正の場合は、見出しの下の空きです。負の場合は、絶対値が見出しの右の空きです (見出しと同じ行から本文を始めます)。
- スタイル** 見出しの文字スタイルの設定です。
- *** この * 印がないと、見出し番号を付け、見出し番号のカウンタに 1 を加算します。
- 別見出し** 目次や柱に出力する見出しです。
- 見出し** 見出しです。

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

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

```

1000 \def\ltjs@patching@startsection{%
1001   \ExplSyntaxOn
1002   \patchcmd{\@startsection}{%
1003     \@afterindenttrue
1004   }{%
1005     \if@english \@afterindentfalse \else \@afterindenttrue \fi
1006   }{}{}
1007   \patchcmd{\@startsection}{%
1008     \addvspace\@tempskipa
1009   }{%
1010     \ifdim \@tempskipa >\z@
1011       \if@slide\else
1012         \null
1013         \vspace*{-\baselineskip}%
1014       \fi
1015       \vskip\@tempskipa
1016     \fi
1017   }{}{}
1018   \ExplSyntaxOff
1019 }
1020 \ltjs@patching@startsection
1021 \let\ltjs@orig@startsection=\@startsection
1022 \AtBeginDocument{%

```



```

1023 \ifx\@startsection\ltjs@orig@startsection\else\expandafter\ltjs@patching@startsection\fi
1024 \let\ltjs@patching@startsection\undefined
1025 }

```

\@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。

```

1026 \def\@sect#1#2#3#4#5#6[#7]#8{%
1027 \ifnum #2>\c@secnumdepth
1028 \let\@svsec\@empty
1029 \else
1030 \refstepcounter{#1}%
1031 \protected@edef\@svsec{\@secntformat{#1}\relax}%
1032 \fi
1033 % 見出し後の空きを \@tempskipa にセット
1034 \@tempskipa #5\relax
1035 % 条件判断の順序を入れ替えました
1036 \ifdim \@tempskipa<\z@
1037 \def\@svsechd{%
1038 #6{\hskip #3\relax
1039 \@svsec #8}%
1040 \csname #1mark\endcsname{#7}%
1041 \addcontentsline{toc}{#1}{%
1042 \ifnum #2>\c@secnumdepth \else
1043 \protect\numberline{\csname the#1\endcsname}%
1044 \fi
1045 #7}}% 目次にフルネームを載せるなら #8
1046 \else
1047 \begingroup
1048 \interlinepenalty \@M % 下から移動
1049 #6{%
1050 \@hangfrom{\hskip #3\relax\@svsec}%
1051 % \interlinepenalty \@M % 上に移動
1052 #8\@par}%
1053 \endgroup
1054 \csname #1mark\endcsname{#7}%
1055 \addcontentsline{toc}{#1}{%
1056 \ifnum #2>\c@secnumdepth \else
1057 \protect\numberline{\csname the#1\endcsname}%
1058 \fi
1059 #7}% 目次にフルネームを載せるならここは #8
1060 \fi
1061 \@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。

[2011-10-05 LTJ] LuaTeX-ja では \everyparhook は不要なので削除。

[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。

```

1062 \def\xsect#1{%
1063 % 見出しの後ろの空きを \@tempskipa にセット
1064 \@tempskipa #1\relax
1065 % 条件判断の順序を変えました
1066 \ifdim \@tempskipa<\z@
1067   \@nbreakfalse
1068   \global\@noskipsectrue
1069   \everypar{%
1070     \if@noskipsec
1071       \global\@noskipsecfalse
1072       {\setbox\z@\lastbox}%
1073       \clubpenalty\@M
1074       \begingroup \@svsechd \endgroup
1075       \unskip
1076       \@tempskipa #1\relax
1077       \hskip -\@tempskipa\ltjfakeparbegin
1078     \else
1079       \clubpenalty \@clubpenalty
1080       \everypar{}%
1081     \fi}%
1082 \else
1083   \par \nbreak
1084   \vskip \@tempskipa
1085   \@afterheading
1086 \fi
1087 \if@slide
1088   {\vskip\if@twocolumn-5\jsc@mpt\else-6\jsc@mpt\fi
1089   \maybeblue\hrule height0\jsc@mpt depth1\jsc@mpt
1090   \vskip\if@twocolumn 4\jsc@mpt\else 7\jsc@mpt\fi\relax}%
1091 \fi
1092 \par % 2000-12-18
1093 \ignorespaces}
1094 \def\ssect#1#2#3#4#5{%
1095 \@tempskipa #3\relax
1096 \ifdim \@tempskipa<\z@
1097   \def\@svsechd{#4{\hskip #1\relax #5}}%
1098 \else
1099   \begingroup
1100     #4{%
1101       \@hangfrom{\hskip #1}%
1102       \interlinepenalty \@M #5\@par}%
1103   \endgroup
1104 \fi
1105 \xsect{#3}}

```

■柱関係の命令

`\chaptermark` `\...mark` の形の命令を初期化します (第 8 節参照)。`\chaptermark` 以外は L^AT_EX 本体で

`\sectionmark`

`\subsectionmark`

`\subsubsectionmark`

`\paragraphmark`

`\subparagraphmark`

定義済みです。

```
1106 \newcommand*\chaptermark[1]{  
1107 % \newcommand*\sectionmark[1]{  
1108 % \newcommand*\subsectionmark[1]{  
1109 % \newcommand*\subsubsectionmark[1]{  
1110 % \newcommand*\paragraphmark[1]{  
1111 % \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。

```
1112 %<!book&!report>\setcounter{secnumdepth}{3}  
1113 %<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 1114 \newcounter{part}  
1115 %<book|report>\newcounter{chapter}  
\c@subsubsection 1116 %<book|report>\newcounter{section}[chapter]  
1117 %<!book&!report>\newcounter{section}  
\c@paragraph 1118 \newcounter{subsection}[section]  
1119 \newcounter{subsubsection}[subsection]  
1120 \newcounter{paragraph}[subsubsection]  
1121 \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。

```
1122 \renewcommand{\thepart}{\@Roman\c@part}  
1123 %<!book&!report>% \renewcommand{\thesection}{\@arabic\c@section}  
1124 %<!book&!report>\renewcommand{\thesection}{\presectionname\@arabic\c@section\postsectionname}  
1125 %<!book&!report>\renewcommand{\thesubsection}{\@arabic\c@section.\@arabic\c@subsection}  
1126 %<*book|report>  
1127 \renewcommand{\thechapter}{\@arabic\c@chapter}  
1128 \renewcommand{\thesection}{\thechapter.\@arabic\c@section}  
1129 \renewcommand{\thesubsection}{\thesection.\@arabic\c@subsection}  
1130 %</book|report>  
1131 \renewcommand{\thesubsubsection}{%  
1132 \thesubsection.\@arabic\c@subsubsection}
```

```

1133 \renewcommand{\theparagraph}{%
1134   \thesubsubsection.\@arabic\c@paragraph}
1135 \renewcommand{\thesubparagraph}{%
1136   \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。

```

1137 %<book|report>\newcommand{\@chapapp}{\prechaptername}
1138 %<book|report>\newcommand{\@chappos}{\postchaptername}

```

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

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

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

```

1139 %<*book>
1140 \newcommand\frontmatter{%
1141   \pltx@cleartooddpage
1142   \@mainmatterfalse
1143   \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。

```

1144 \newcommand\mainmatter{%
1145   \pltx@cleartooddpage
1146   \@mainmattertrue
1147   \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。

```

1148 \newcommand\backmatter{%
1149   \if@openleft
1150     \cleardoublepage
1151   \else\if@openright
1152     \cleardoublepage
1153   \else
1154     \clearpage
1155   \fi\fi
1156   \@mainmatterfalse}
1157 %</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。

```
1158 %<!*book&!report>
1159 \newcommand\part{%
1160   \if@noskipsec \leavevmode \fi
1161   \par
1162   \addvspace{4ex}%
1163   \if@english \@afterindentfalse \else \@afterindenttrue \fi
1164   \secdef\@part\@spart}
1165 %</!*book&!report>
```

`book` および `report` クラスの場合は、少し複雑です。

```
1166 %<*book|report>
1167 \newcommand\part{%
1168   \if@openleft
1169     \cleardoublepage
1170   \else\if@openright
1171     \cleardoublepage
1172   \else
1173     \clearpage
1174   \fi\fi
1175   \thispagestyle{empty}% 欧文用標準スタイルでは plain
1176   \if@twocolumn
1177     \onecolumn
1178     \@restonecoltrue
1179   \else
1180     \@restonecolfalse
1181   \fi
1182   \null\vfil
1183   \secdef\@part\@spart}
1184 %</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。

```
1185 %<!*book&!report>
```

```

1186 \def\@part [#1]#2{%
1187   \ifnum \c@secnumdepth >\m@ne
1188     \refstepcounter{part}%
1189     \addcontentsline{toc}{part}{%
1190       \prepartname\thepart\postpartname\hspace{1\zw}#1}%
1191   \else
1192     \addcontentsline{toc}{part}{#1}%
1193   \fi
1194   \markboth{}{}%
1195   {\parindent\z@
1196     \raggedright
1197     \interlinepenalty \@M
1198     \normalfont
1199     \ifnum \c@secnumdepth >\m@ne
1200       \Large\headfont\prepartname\thepart\postpartname
1201       \par\nobreak
1202     \fi
1203     \huge \headfont #2%
1204     \markboth{}{}\par}%
1205   \nobreak
1206   \vskip 3ex
1207   \@afterheading}
1208 %<!/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。

```

1209 %<*book|report>
1210 \def\@part [#1]#2{%
1211   \ifnum \c@secnumdepth >-2\relax
1212     \refstepcounter{part}%
1213     \addcontentsline{toc}{part}{%
1214       \prepartname\thepart\postpartname\hspace{1\zw}#1}%
1215   \else
1216     \addcontentsline{toc}{part}{#1}%
1217   \fi
1218   \markboth{}{}%
1219   {\centering
1220     \interlinepenalty \@M
1221     \normalfont
1222     \ifnum \c@secnumdepth >-2\relax
1223       \huge\headfont \prepartname\thepart\postpartname
1224       \par\vskip20\jsc@mpt
1225     \fi
1226     \Huge \headfont #2\par}%
1227   \@endpart}
1228 %</book|report>

```

\@spart 番号を付けない部です。

```

1229 %<*!book&!report>
1230 \def\@spart#1{%

```

```

1231 \parindent \z@ \raggedright
1232 \interlinepenalty \@M
1233 \normalfont
1234 \huge \headfont #1\par}%
1235 \nobreak
1236 \vskip 3ex
1237 \@afterheading}
1238 %<!/book&!report>
1239 %<*book|report>
1240 \def\@spart#1{%
1241 \centering
1242 \interlinepenalty \@M
1243 \normalfont
1244 \Huge \headfont #1\par}%
1245 \@endpart}
1246 %</book|report>

```

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

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

```

1247 %<*book|report>
1248 \def\@endpart{\vfil\newpage
1249 \if@twoside
1250 \if@openleft %% added (2017/02/24)
1251 \null\thispagestyle{empty}\newpage
1252 \else\if@openright %% added (2016/12/13)
1253 \null\thispagestyle{empty}\newpage
1254 \fi\fi %% added (2016/12/13, 2017/02/24)
1255 \fi
1256 \if@restonecol
1257 \twocolumn
1258 \fi}
1259 %</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。

```

1260 %<*book|report>
1261 \newcommand{\chapter}{%
1262 \if@openleft\cleardoublepage\else
1263 \if@openright\cleardoublepage\else\clearpage\fi\fi
1264 \plainifnotempty % 元: \thispagestyle{plain}
1265 \global\@topnum\z@
1266 \if@english \@afterindentfalse \else \@afterindenttrue \fi

```

```

1267 \secdef
1268   {\@omit@numberfalse\@chapter}%
1269   {\@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。

```

1270 \def\@chapter[#1]#2{%
1271   \ifnum \c@secnumdepth >\m@ne
1272 %<book>   \if@mainmatter
1273     \refstepcounter{chapter}%
1274     \typeout{\@chapapp\thechapter\@chappos}%
1275     \addcontentsline{toc}{chapter}%
1276       {\protect\numberline
1277        % {\if@english\thechapter\else\@chapapp\thechapter\@chappos\fi}%
1278        {\@chapapp\thechapter\@chappos}%
1279        #1}%
1280 %<book>   \else\addcontentsline{toc}{chapter}{#1}\fi
1281   \else
1282     \addcontentsline{toc}{chapter}{#1}%
1283   \fi
1284   \chaptermark{#1}%
1285   \addtocontents{lof}{\protect\advspace{10\jsc@empt}}%
1286   \addtocontents{lot}{\protect\advspace{10\jsc@empt}}%
1287   \if@twocolumn
1288     \topnewpage[\@makechapterhead{#2}]%
1289   \else
1290     \@makechapterhead{#2}%
1291     \@afterheading
1292   \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。

```

1293 \def\@makechapterhead#1{%
1294   \vspace*{2\Cvs}% 欧文は 50pt
1295   {\parindent \z@ \raggedright \normalfont
1296    \ifnum \c@secnumdepth >\m@ne
1297 %<book>   \if@mainmatter
1298     \huge\headfont \@chapapp\thechapter\@chappos
1299     \par\nobreak
1300     \vskip \Cvs % 欧文は 20pt
1301 %<book>   \fi
1302   \fi
1303   \interlinepenalty\@M
1304   \Huge \headfont #1\par\nobreak
1305   \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。

```

1306 \def\@schapter#1{%
1307   \chaptermark{#1}%
1308   \if@twocolumn

```



```

1309 \topnewpage[\@makeschapterhead{#1}]%
1310 \else
1311 \@makeschapterhead{#1}\@afterheading
1312 \fi}

```

\@makeschapterhead 番号なしの章見出しです。

```

1313 \def\@makeschapterhead#1{%
1314 \vspace*{2\Cvs}% 欧文は 50pt
1315 {\parindent \z@ \raggedright
1316 \normalfont
1317 \interlinepenalty\@M
1318 \Huge \headfont #1\par\nobreak
1319 \vskip 3\Cvs}} % 欧文は 40pt
1320 %</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。

```

1321 \if@twocolumn
1322 \newcommand{\section}{%
1323 %<jspf>\ifx\maketitle\relax\else\maketitle\fi
1324 \@startsection{section}{1}{\z@}%
1325 %<kiyou> {0.6\Cvs}{0.4\Cvs}%
1326 %<kiyou> {\Cvs}{0.5\Cvs}%
1327 % {\normalfont\large\headfont\@secapp}}
1328 {\normalfont\large\headfont\raggedright}}
1329 \else
1330 \newcommand{\section}{%
1331 \if@slide\clearpage\fi
1332 \@startsection{section}{1}{\z@}%
1333 {\Cvs \@plus.5\Cdp \@minus.2\Cdp}% 前アキ
1334 {.5\Cvs \@plus.3\Cdp}% 後アキ
1335 % {\normalfont\Large\headfont\@secapp}}
1336 {\normalfont\Large\headfont\raggedright}}
1337 \fi

```

\subsection 同上です。

```

1338 \if@twocolumn
1339 \newcommand{\subsection}{\@startsection{subsection}{2}{\z@}%
1340 {\z@}{\if@slide .4\Cvs \else \z@ \fi}%
1341 {\normalfont\normalsize\headfont}}
1342 \else
1343 \newcommand{\subsection}{\@startsection{subsection}{2}{\z@}%
1344 {\Cvs \@plus.5\Cdp \@minus.2\Cdp}% 前アキ
1345 {.5\Cvs \@plus.3\Cdp}% 後アキ
1346 {\normalfont\large\headfont}}

```

1347 \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)。

1348 \if@twocolumn

1349 \newcommand{\subsubsection}{\@startsection{subsubsection}{3}{\z@}%

1350 {\z@}{\if@slide .4\Cvs \else \z@ \fi}%

1351 {\normalfont\normalsize\headfont}}

1352 \else

1353 \newcommand{\subsubsection}{\@startsection{subsubsection}{3}{\z@}%

1354 {\Cvs \@plus.5\Cdp \@minus.2\Cdp}%

1355 {\if@slide .5\Cvs \@plus.3\Cdp \else \z@ \fi}%

1356 {\normalfont\normalsize\headfont}}

1357 \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。

1358 %<!jspf>\newcommand{\jsParagraphMark}{■}

1359 \if@twocolumn

1360 \newcommand{\paragraph}{\@startsection{paragraph}{4}{\z@}%

1361 {\z@}{\if@slide .4\Cvs \else -1\zw\fi}% 改行せず 1\zw のアキ

1362 %<jspf> {\normalfont\normalsize\headfont}}

1363 %<!jspf> {\normalfont\normalsize\headfont\jsParagraphMark}}

1364 \else

1365 \newcommand{\paragraph}{\@startsection{paragraph}{4}{\z@}%

1366 {0.5\Cvs \@plus.5\Cdp \@minus.2\Cdp}%

1367 {\if@slide .5\Cvs \@plus.3\Cdp \else -1\zw\fi}% 改行せず 1\zw のアキ

1368 %<jspf> {\normalfont\normalsize\headfont}}

1369 %<!jspf> {\normalfont\normalsize\headfont\jsParagraphMark}}

1370 \fi

\subparagraph 見出しの後ろで改行されません。

1371 \if@twocolumn

1372 \newcommand{\subparagraph}{\@startsection{subparagraph}{5}{\z@}%

1373 {\z@}{\if@slide .4\Cvs \@plus.3\Cdp \else -1\zw\fi}%

1374 {\normalfont\normalsize\headfont}}

1375 \else

1376 \newcommand{\subparagraph}{\@startsection{subparagraph}{5}{\z@}%

1377 {\z@}{\if@slide .5\Cvs \@plus.3\Cdp \else -1\zw\fi}%

1378 {\normalfont\normalsize\headfont}}

1379 \fi

9.3 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] `3\zw` に変更しました。

[2005-03-19] 二段組は `2\zw` に戻しました。

```
1380 \if@slide
1381   \setlength\leftmargini{1\zw}
1382 \else
1383   \if@twocolumn
1384     \setlength\leftmargini{2\zw}
1385   \else
1386     \setlength\leftmargini{3\zw}
1387   \fi
1388 \fi
```

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

```
\leftmarginiv 1389 \if@slide
\leftmarginv 1390   \setlength\leftmarginii {1\zw}
\leftmarginvi 1391   \setlength\leftmarginiii{1\zw}
1392   \setlength\leftmarginiv {1\zw}
1393   \setlength\leftmarginv  {1\zw}
1394   \setlength\leftmarginvi {1\zw}
1395 \else
1396   \setlength\leftmarginii {2\zw}
1397   \setlength\leftmarginiii{2\zw}
1398   \setlength\leftmarginiv {2\zw}
1399   \setlength\leftmarginv  {1\zw}
1400   \setlength\leftmarginvi {1\zw}
1401 \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。

```
1402 \setlength \labelsep {0.5\zw} % .5em
1403 \setlength \labelwidth{\leftmargini}
1404 \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。

```
1405 \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`

`\@itempenalty`

```

1406 \@beginparpenalty -\@lowpenalty
1407 \@endparpenalty -\@lowpenalty
1408 \@itempenalty -\@lowpenalty

```

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

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

```

1409 \def\@listi{\leftmargin\leftmarginI
1410   \parsep \z@
1411   \topsep 0.5\baselineskip
1412   \itemsep \z@ \relax}
1413 \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）。

```
1414 \@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 1415 \def\@listii{\leftmargin\leftmarginii
\@listiv 1416   \labelwidth\leftmarginii \advance\labelwidth-\labelsep
1417   \topsep \z@
\@listv 1418   \parsep \z@
\@listvi 1419   \itemsep\parsep}
1420 \def\@listiii{\leftmargin\leftmarginiii
1421   \labelwidth\leftmarginiii \advance\labelwidth-\labelsep
1422   \topsep \z@
1423   \parsep \z@
1424   \itemsep\parsep}
1425 \def\@listiv {\leftmargin\leftmarginiv
1426   \labelwidth\leftmarginiv
1427   \advance\labelwidth-\labelsep}
1428 \def\@listv {\leftmargin\leftmarginv
1429   \labelwidth\leftmarginv
1430   \advance\labelwidth-\labelsep}
1431 \def\@listvi {\leftmargin\leftmarginvi
1432   \labelwidth\leftmarginvi
1433   \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 `\theenumiii`
`\theenumiv`

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。

```
1434 \renewcommand{\theenumi}{\@arabic\c@enumi}
1435 \renewcommand{\theenumii}{\@alph\c@enumii}
1436 \renewcommand{\theenumiii}{\@roman\c@enumiii}
1437 \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 `\labelenumiii` 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 1438 \newcommand{\labelenumi}{\theenumi.}
1439 \newcommand{\labelenumii}{\inhibitglue (\theenumii) \inhibitglue}
1440 \newcommand{\labelenumiii}{\theenumiii.}
1441 \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 1442 \renewcommand{\p@enumii}{\theenumi}
1443 \renewcommand{\p@enumiii}{\theenumi\inhibitglue (\theenumii) }
1444 \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 1445 \newcommand\labelitemi{\textbullet}
\labelitemiii 1446 \newcommand\labelitemii{\normalfont\bfseries \textendash}
1447 \newcommand\labelitemiii{\textasteriskcentered}
\labelitemiv 1448 \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。

```
1449 \newenvironment{description}{%
1450 \list{}{%
1451 \labelwidth=\leftmargin
1452 \labelsep=1\zw
1453 \advance \labelwidth by -\labelsep
1454 \let \makelabel=\descriptionlabel}}{\endlist}
```

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

```
1455 \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。

```

1456 %<*book>
1457 \newenvironment{abstract}{%
1458   \begin{list}{}{%
1459     \listparindent=1\zw
1460     \itemindent=\listparindent
1461     \rightmargin=0pt
1462     \leftmargin=5\zw}\item[]}{\end{list}\vspace{\baselineskip}}
1463 %</book>
1464 %<*article|report|kiyou>
1465 \newbox\@abstractbox
1466 \if@titlepage
1467   \newenvironment{abstract}{%
1468     \titlepage
1469     \null\vfil
1470     \@beginparpenalty\@lowpenalty
1471     \begin{center}%
1472       \headfont \abstractname
1473       \@endparpenalty\@M
1474     \end{center}}%
1475   {\par\vfil\null\endtitlepage}
1476 \else
1477   \newenvironment{abstract}{%
1478     \if@twocolumn
1479       \ifx\maketitle\relax
1480         \section*{\abstractname}%
1481       \else
1482         \global\setbox\@abstractbox\hbox\bgroup
1483         \begin{minipage}[b]{\textwidth}
1484           \small\parindent1\zw
1485           \begin{center}%
1486             {\headfont \abstractname\vspace{-.5em}\vspace{\z@}}%
1487           \end{center}%
1488           \list{}{%
1489             \listparindent\parindent
1490             \itemindent \listparindent
1491             \rightmargin \leftmargin}%
1492           \item\relax
1493         \fi
1494       \else
1495         \small
1496         \begin{center}%
1497           {\headfont \abstractname\vspace{-.5em}\vspace{\z@}}%

```

```

1498     \end{center}%
1499     \list{}{%
1500         \listparindent\parindent
1501         \itemindent \listparindent
1502         \rightmargin \leftmargin}%
1503     \item\relax
1504 \fi}{\if@twocolumn
1505     \ifx\maketitle\relax
1506     \else
1507         \endlist\end{minipage}\egroup
1508     \fi
1509 \else
1510     \endlist
1511 \fi}
1512 \fi
1513 %</article|report|kiyou>
1514 %<*jspf>
1515 \newbox\@abstractbox
1516 \newenvironment{abstract}{%
1517     \global\setbox\@abstractbox\hbox\bgroup
1518     \begin{minipage}[b]{157\jsc@mmm}{\sffamily Abstract}\par
1519     \small
1520     \if@english \parindent6\jsc@mmm \else \parindent1\zw \fi}%
1521     {\end{minipage}\egroup}
1522 %</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。

```

1523 %<*jspf>
1524 %\newbox\@keywordsbox
1525 %\newenvironment{keywords}{%
1526 %     \global\setbox\@keywordsbox\hbox\bgroup
1527 %     \begin{minipage}[b]{157\jsc@mmm}{\sffamily Keywords:}\par
1528 %     \small\parindent0\zw}%
1529 %     {\end{minipage}\egroup}
1530 %</jspf>

```

■verse 環境

`verse (env.)` 詩のための `verse` 環境です。

```

1531 \newenvironment{verse}{%
1532     \let \=\@centercr
1533     \list{}{%
1534         \itemsep \z@
1535         \itemindent -2\zw % 元: -1.5em
1536         \listparindent\itemindent
1537         \rightmargin \z@

```

```
1538 \advance\leftmargin 2\zw}% 元: 1.5em
1539 \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。

```
1540 \newenvironment{quotation}{%
1541 \list{}{%
1542 \listparindent\parindent
1543 \itemindent\listparindent
1544 \rightmargin \z@}%
1545 \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。

```
1546 \newenvironment{quote}%
1547 {\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 `1\zw` にし、括弧を全角にしました。

```
1548 \def\@begintheorem#1#2{\trivlist\labelsep=1\zw
1549 \item[\hskip \labelsep{\headfont #1\ #2}]}
1550 \def\@opargbegintheorem#1#2#3{\trivlist\labelsep=1\zw
1551 \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。

```
1552 \newenvironment{titlepage}{%
1553 %<book> \pltx@cleartooddpage %% 2017-02-24
1554 \if@twocolumn
1555 \@restonecoltrue\onecolumn
```



```

1556 \else
1557 \@restonecolfalse\newpage
1558 \fi
1559 \thispagestyle{empty}%
1560 \ifodd\c@page\setcounter{page}\@ne\else\setcounter{page}\z@\fi
1561 }%
1562 {\if@restonecol\twocolumn \else \newpage \fi
1563 \if@twoside\else
1564 \setcounter{page}\@ne
1565 \fi}

```

■付録

`\appendix` 本文と付録を分離するコマンドです。

```

1566 %<!*book&!report>
1567 \newcommand{\appendix}{\par
1568 \setcounter{section}{0}%
1569 \setcounter{subsection}{0}%
1570 \gdef\presectionname{\appendixname}%
1571 \gdef\postsectionname{}}
1572 % \gdef\thesection{\@Alph\c@section}% [2003-03-02]
1573 \gdef\thesection{\presectionname\@Alph\c@section\postsectionname}%
1574 \gdef\thesubsection{\@Alph\c@section.\@arabic\c@subsection}}
1575 %</!*book&!report>
1576 %<*book|report>
1577 \newcommand{\appendix}{\par
1578 \setcounter{chapter}{0}%
1579 \setcounter{section}{0}%
1580 \gdef\@chapapp{\appendixname}%
1581 \gdef\@chappos{}}
1582 \gdef\thechapter{\@Alph\c@chapter}}
1583 %</book|report>

```

9.4 パラメータの設定

■array と tabular 環境

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

```
1584 \setlength\arraycolsep{5\jsc@empt}
```

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

```
1585 \setlength\tabcolsep{6\jsc@empt}
```

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

```
1586 \setlength\arrayrulewidth{.4\jsc@empt}
```

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

```
1587 \setlength\doublerulesep{2\jsc@empt}
```

■tabbing 環境

`\tabbingsep` `\'` コマンドで入るアキです。

```
1588 \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。

```
1589 \skip\@mpfootins = \skip\footins
```

■framebox 環境

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

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

```
1590 \setlength\fboxsep{3\jsc@mp}
```

```
1591 \setlength\fboxrule{.4\jsc@mp}
```

■equation と eqnarray 環境

`\theequation` 数式番号を出力するコマンドです。

```
1592 %<!book&!report>\renewcommand \theequation {\@arabic\c@equation}
```

```
1593 %<*book|report>
```

```
1594 \@addtoreset{equation}{chapter}
```

```
1595 \renewcommand\theequation
```

```
1596 {\ifnum \c@chapter>\z@ \thechapter.\fi \@arabic\c@equation}
```

```
1597 %</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。

```
1598 % \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。

```
1599 % \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。

```
1600 % \def\tagform@#1{\maketag@@@{ (\ignorespaces#1\unskip\@@italiccorr ) }}
```

9.5 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。

```
1601 %<!*book&!report>
1602 \newcounter{figure}
1603 \renewcommand \thefigure {\@arabic\c@figure}
1604 %</!*book&!report>
1605 %<*book|report>
1606 \newcounter{figure}[chapter]
1607 \renewcommand \thefigure
1608     {\ifnum \c@chapter>\z@ \thechapter.\fi \@arabic\c@figure}
1609 %</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 1610 \def\fps@figure{tbp}
1611 \def\ftype@figure{1}
\fnum@figure 1612 \def\ext@figure{lof}
1613 \def\fnum@figure{\figurename\nobreak\thefigure}
```

`figure (env.)` * 形式は段抜きのフロートです。

```
figure* (env.) 1614 \newenvironment{figure}%
1615     {\@float{figure}}%
1616     {\end@float}
1617 \newenvironment{figure*}%
1618     {\@dblfloat{figure}}%
1619     {\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。

```
1620 %<!*book&!report>
1621 \newcounter{table}
1622 \renewcommand\thetable{\@arabic\c@table}
1623 %</!*book&!report>
1624 %<*book|report>
1625 \newcounter{table}[chapter]
```

```

1626 \renewcommand \thetable
1627     {\ifnum \c@chapter>\z@ \thechapter.\fi \@arabic\c@table}
1628 %</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 1629 \def\fps@table{tbp}
\fnun@table 1630 \def\ftype@table{2}
1631 \def\ext@table{lot}
1632 \def\fnun@table{\tablename\nobreak\thetable}

```

`table (env.) *` は段抜きのフロートです。

```

table* (env.) 1633 \newenvironment{table}%
1634             {\@float{table}}%
1635             {\end@float}
1636 \newenvironment{table*}%
1637             {\@dblfloat{table}}%
1638             {\end@dblfloat}

```

9.6 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` ていたので、キャプションを表の上につけた場合にキャプションと表がくっついてしま
 うのを直しました。

```

1639 \newlength\abovecaptionskip
1640 \newlength\belowcaptionskip
1641 \setlength\abovecaptionskip{5\jsc@mp} % 元: 10\p@
1642 \setlength\belowcaptionskip{5\jsc@mp} % 元: 0\p@

```

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

[2003-11-05] ロジックを少し変えてみました。

[2015-05-26] `listings` パッケージを使うときに `title` を指定すると次のエラーが出るの
 を修正。

```
! Missing number, treated as zero.
```

```

1643 %<*\jspf>
1644 % \long\def\@makecaption#1#2{\small
1645 %   \advance\leftskip10\jsc@mmm
1646 %   \advance\rightskip10\jsc@mmm
1647 %   \vskip\abovecaptionskip
1648 %   \sbox\@tempboxa{#1{\hskip1\zw}#2}%

```

```

1649 % \ifdim \wd\@tempboxa >\hsize
1650 % #1{\hskip1\zw}#2\par
1651 % \else
1652 % \global \@minipagefalse
1653 % \hb@xt@\hsize{\hfil\box\@tempboxa\hfil}%
1654 % \fi
1655 % \vskip\belowcaptionskip}}
1656 \long\def\@makecaption#1#2{\small
1657 \advance\leftskip .0628\linewidth
1658 \advance\rightskip .0628\linewidth
1659 \vskip\abovecaptionskip
1660 \sbox\@tempboxa{#1{\hskip1\zw}#2}%
1661 \ifdim \wd\@tempboxa <\hsize \centering \fi
1662 #1{\hskip1\zw}#2\par
1663 \vskip\belowcaptionskip}}
1664 %</!jspf>
1665 %<*jspf>
1666 \long\def\@makecaption#1#2{%
1667 \vskip\abovecaptionskip
1668 \sbox\@tempboxa{\small\sffamily #1\quad #2}%
1669 \ifdim \wd\@tempboxa >\hsize
1670 {\small\sffamily
1671 \list{#1}{%
1672 \renewcommand{\makelabel}[1]{##1\hfil}
1673 \itemsep \z@
1674 \itemindent \z@
1675 \labelsep \z@
1676 \labelwidth 11\jsc@mmm
1677 \listparindent\z@
1678 \leftmargin 11\jsc@mmm}\item\relax #2\endlist}
1679 \else
1680 \global \@minipagefalse
1681 \hb@xt@\hsize{\hfil\box\@tempboxa\hfil}%
1682 \fi
1683 \vskip\belowcaptionskip}
1684 %</jspf>

```

10 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。

[2016-07-15] KOMA-Script 中の `\scr@DeclareOldFontCommand` に倣い、これらの命令を使うときには警告を発することにしました。

[2016-07-16] 警告を最初の一回だけ発することにしました。また、例外的に警告を出さないようにするスイッチも付けます。

```

\if@jsc@warnoldfontcmd
\if@jsc@warnoldfontcmdexception 1685 \newif\if@jsc@warnoldfontcmd
1686 \@jsc@warnoldfontcmdtrue
1687 \newif\if@jsc@warnoldfontcmdexception
1688 \@jsc@warnoldfontcmdexceptionfalse

\jsc@DeclareOldFontCommand
1689 \newcommand*{\jsc@DeclareOldFontCommand}[3]{%
1690   \DeclareOldFontCommand{#1}{%
1691     \jsc@warnoldfontcmd{#1}#2%
1692   }{%
1693     \jsc@warnoldfontcmd{#1}#3%
1694   }%
1695 }
1696 \DeclareRobustCommand*{\jsc@warnoldfontcmd}[1]{%
1697   \if@jsc@warnoldfontcmdexception\else\if@jsc@warnoldfontcmd
1698     \ClassWarning{\jsc@clsname}{%
1699       deprecated old font command `string#1' used.\MessageBreak
1700       You should note, that since 1994 LaTeX2e provides a\MessageBreak
1701       new font selection scheme called NFSS2 with several\MessageBreak
1702       new, combinable font commands. This \jsc@clsname\MessageBreak
1703       class has defined the old font commands like\MessageBreak
1704       `string#1' only for compatibility%
1705     }%
1706     \global\@jsc@warnoldfontcmdfalse
1707     \fi\fi
1708 }

\mc フォントファミリーを変更します。
\gt 1709 \jsc@DeclareOldFontCommand{\mc}{\normalfont\mcfamily}{\mathmc}
\rm 1710 \jsc@DeclareOldFontCommand{\gt}{\normalfont\gtfamily}{\mathgt}
1711 \jsc@DeclareOldFontCommand{\rm}{\normalfont\rmfamily}{\mathrm}
\sf 1712 \jsc@DeclareOldFontCommand{\sf}{\normalfont\sffamily}{\mathsf}
\tt 1713 \jsc@DeclareOldFontCommand{\tt}{\normalfont\ttfamily}{\mathtt}

\bf ボールドシリーズにします。通常のミディアムシリーズに戻すコマンドは \mdseries
です。
1714 \jsc@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
\s1 せん（警告メッセージを出力します）。通常のアップライト体に戻すコマンドは \upshape
\sc です。
1715 \jsc@DeclareOldFontCommand{\it}{\normalfont\itshape}{\mathit}
1716 \jsc@DeclareOldFontCommand{\s1}{\normalfont\s1shape}{\@nomath\s1}
1717 \jsc@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 1718 \DeclareRobustCommand*{\cal}{\@fontswitch\relax\mathcal}
1719 \DeclareRobustCommand*{\mit}{\@fontswitch\relax\mathnormal}

```

11 相互参照

11.1 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。

```
1720 \newcommand\@pnumwidth{1.55em}
```

```
1721 \newcommand\@tocrmarg{2.55em}
```

```
1722 \newcommand\@dotsep{4.5}
```

```
1723 %<!book&!report>\setcounter{tocdepth}{2}
```

```
1724 %<book|report>\setcounter{tocdepth}{1}
```

■目次

`\tableofcontents` 目次を生成します。

`\jsc@tocl@width` [2013-12-30] `\prechaptername` などから見積もった目次のラベルの長さです。(by ts)

```
1725 \newdimen\jsc@tocl@width
1726 \newcommand{\tableofcontents}{%
1727 %<*book|report>
1728 \settowidth\jsc@tocl@width{\headfont\prechaptername\postchaptername}%
1729 \settowidth\@tempdima{\headfont\appendixname}%
1730 \ifdim\jsc@tocl@width<\@tempdima \setlength\jsc@tocl@width{\@tempdima}\fi
1731 \ifdim\jsc@tocl@width<2\zw \divide\jsc@tocl@width by 2 \advance\jsc@tocl@width 1\zw\fi
1732 \if@twocolumn
1733 \@restonecoltrue\onecolumn
1734 \else
1735 \@restonecolfalse
1736 \fi
1737 \chapter*{\contentsname}%
1738 \@mkboth{\contentsname}{}%
1739 %</book|report>
1740 %<!*book&!report>
1741 \settowidth\jsc@tocl@width{\headfont\presectionname\postsectionname}%
1742 \settowidth\@tempdima{\headfont\appendixname}%
1743 \ifdim\jsc@tocl@width<\@tempdima\relax\setlength\jsc@tocl@width{\@tempdima}\fi
1744 \ifdim\jsc@tocl@width<2\zw \divide\jsc@tocl@width by 2 \advance\jsc@tocl@width 1\zw\fi
1745 \section*{\contentsname}%
1746 \@mkboth{\contentsname}{\contentsname}%
1747 %</!book&!report>
1748 \@starttoc{toc}%
1749 %<book|report> \if@restonecol\twocolumn\fi
1750 }
```

`\l@part` 部の目次です。

```
1751 \newcommand*{\l@part}[2]{%
1752 \ifnum \c@tocdepth >-2\relax
1753 %<!book&!report> \addpenalty\@secpenalty
1754 %<book|report> \addpenalty{-\@highpenalty}%
1755 \addvspace{2.25em \@plus\jsc@empt}%
1756 \begingroup
1757 \parindent \z@
1758 % \@pnumwidth should be \@tocrmarg
1759 % \rightskip \@pnumwidth
1760 \rightskip \@tocrmarg
1761 \parfillskip -\rightskip
1762 {\leavevmode
1763 \large \headfont
1764 \setlength\@lnumwidth{4\zw}%
1765 #1\hfil \hb@xt@\@pnumwidth{\hss #2}}\par
1766 \nobreak
1767 %<book|report> \global\@nobreaktrue
1768 %<book|report> \everypar{\global\@nobreakfalse\everypar{}}%
```



```
1769 \endgroup
1770 \fi}
```

`\l@chapter` 章の目次です。 `\@lnumwidth` を `4.683\zw` 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)

```
1771 %<*book|report>
1772 \newcommand*{\l@chapter}[2]{%
1773 \ifnum \c@tocdepth >\m@ne
1774 \addpenalty{-\@highpenalty}%
1775 \addvspace{1.0em \@plus\jsc@mp}
1776 % \vskip 1.0em \@plus\p@ % book.cls では↑がこうなっている
1777 \begingroup
1778 \parindent\z@
1779 % \rightskip\@pnumwidth
1780 \rightskip\@tocmarg
1781 \parfillskip-\rightskip
1782 \leavevmode\headfont
1783 % \if@english\setlength\@lnumwidth{5.5em}\else\setlength\@lnumwidth{4.683\zw}\fi
1784 \setlength\@lnumwidth{\jsc@tocl@width}\advance\@lnumwidth 2.683\zw
1785 \advance\leftskip\@lnumwidth \hskip-\leftskip
1786 #1\nobreak\hfil\nobreak\hbox to\@pnumwidth{\hss#2}\par
1787 \penalty\@highpenalty
1788 \endgroup
1789 \fi}
1790 %</book|report>
```

`\l@section` 節の目次です。

```
1791 %<!*book&!report>
1792 \newcommand*{\l@section}[2]{%
1793 \ifnum \c@tocdepth >\z@
1794 \addpenalty{\@secpenalty}%
1795 \addvspace{1.0em \@plus\jsc@mp}%
1796 \begingroup
1797 \parindent\z@
1798 % \rightskip\@pnumwidth
1799 \rightskip\@tocmarg
1800 \parfillskip-\rightskip
1801 \leavevmode\headfont
1802 %\setlength\@lnumwidth{4\zw}% 元 1.5em [2003-03-02]
1803 \setlength\@lnumwidth{\jsc@tocl@width}\advance\@lnumwidth 2\zw
1804 \advance\leftskip\@lnumwidth \hskip-\leftskip
1805 #1\nobreak\hfil\nobreak\hbox to\@pnumwidth{\hss#2}\par
1806 \endgroup
1807 \fi}
1808 %</!book&!report>
```

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

```
1809 %<book|report> % \newcommand*{\l@section}{\dottedtocline{1}{1\zw}{3.683\zw}}
```

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

\l@section 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@subsection しれません。

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

```
\l@subparagraph 1810 %<!*book&!report>
1811 % \newcommand*\l@subsection {\@dottedtocline{2}{1.5em}{2.3em}}
1812 % \newcommand*\l@subsubsection{\@dottedtocline{3}{3.8em}{3.2em}}
1813 % \newcommand*\l@paragraph {\@dottedtocline{4}{7.0em}{4.1em}}
1814 % \newcommand*\l@subparagraph {\@dottedtocline{5}{10em}{5em}}
1815 %
1816 % \newcommand*\l@subsection {\@dottedtocline{2}{1\zw}{3\zw}}
1817 % \newcommand*\l@subsubsection{\@dottedtocline{3}{2\zw}{3\zw}}
1818 % \newcommand*\l@paragraph {\@dottedtocline{4}{3\zw}{3\zw}}
1819 % \newcommand*\l@subparagraph {\@dottedtocline{5}{4\zw}{3\zw}}
1820 %
1821 \newcommand*\l@subsection}{%
1822     \@tempdima\jsc@tocl@width \advance\@tempdima -1\zw
1823     \@dottedtocline{2}{\@tempdima}{3\zw}}
1824 \newcommand*\l@subsubsection}{%
1825     \@tempdima\jsc@tocl@width \advance\@tempdima 0\zw
1826     \@dottedtocline{3}{\@tempdima}{4\zw}}
1827 \newcommand*\l@paragraph}{%
1828     \@tempdima\jsc@tocl@width \advance\@tempdima 1\zw
1829     \@dottedtocline{4}{\@tempdima}{5\zw}}
1830 \newcommand*\l@subparagraph}{%
1831     \@tempdima\jsc@tocl@width \advance\@tempdima 2\zw
1832     \@dottedtocline{5}{\@tempdima}{6\zw}}
1833 %</!*book&!report>
1834 %<*book|report>
1835 % \newcommand*\l@subsection {\@dottedtocline{2}{3.8em}{3.2em}}
1836 % \newcommand*\l@subsubsection{\@dottedtocline{3}{7.0em}{4.1em}}
1837 % \newcommand*\l@paragraph {\@dottedtocline{4}{10em}{5em}}
1838 % \newcommand*\l@subparagraph {\@dottedtocline{5}{12em}{6em}}
1839 \newcommand*\l@section}{%
1840     \@tempdima\jsc@tocl@width \advance\@tempdima -1\zw
1841     \@dottedtocline{1}{\@tempdima}{3.683\zw}}
1842 \newcommand*\l@subsection}{%
1843     \@tempdima\jsc@tocl@width \advance\@tempdima 2.683\zw
1844     \@dottedtocline{2}{\@tempdima}{3.5\zw}}
1845 \newcommand*\l@subsubsection}{%
1846     \@tempdima\jsc@tocl@width \advance\@tempdima 6.183\zw
1847     \@dottedtocline{3}{\@tempdima}{4.5\zw}}
1848 \newcommand*\l@paragraph}{%
1849     \@tempdima\jsc@tocl@width \advance\@tempdima 10.683\zw
1850     \@dottedtocline{4}{\@tempdima}{5.5\zw}}
1851 \newcommand*\l@subparagraph}{%
1852     \@tempdima\jsc@tocl@width \advance\@tempdima 16.183\zw
```

```
1853 \dottedtocline{5}{\tempdima}{6.5\zw}}
1854 %</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。

```
1855 \newdimen\@lnumwidth
1856 \def\numberline#1{\hb@xt@\@lnumwidth{#1\hfil}\hspace{0pt}}
```

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

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

```
1857 \def\jsTocLine{\leaders\hbox{%
1858 $ \m@th \mkern \@dotsep mu\hbox{.}\mkern \@dotsep mu$}\hfill}
1859 \def\dottedtocline#1#2#3#4#5{\ifnum #1>\c@tocdepth \else
1860 \vskip \z@ \@plus.2\jcs@mp
1861 {\leftskip #2\relax \rightskip \@tocrmarg \parfillskip -\rightskip
1862 \parindent #2\relax\@afterindenttrue
1863 \interlinepenalty\@M
1864 \leavevmode
1865 \@lnumwidth #3\relax
1866 \advance\leftskip \@lnumwidth \null\nobreak\hskip -\leftskip
1867 {#4}\nobreak
1868 \jsTocLine \nobreak\hb@xt@\@pnumwidth{%
1869 \hfil\normalfont \normalcolor #5}\par}\fi}
```

■ 図目次と表目次

`\listoffigures` 図目次を出力します。

```
1870 \newcommand{\listoffigures}{%
1871 %<*book|report>
1872 \if@twocolumn\@restonecoltrue\onecolumn
1873 \else\@restonecolfalse\fi
1874 \chapter*{\listfigurename}%
1875 \mkboth{\listfigurename}{}%
1876 %</book|report>
1877 %<!*book&!report>
1878 \section*{\listfigurename}%
1879 \mkboth{\listfigurename}{\listfigurename}%
1880 %</!*book&!report>
1881 \@starttoc{lof}%
```

```
1882 %<book|report> \if@restonecol\twocolumn\fi
1883 }
```

`\l@figure` 図目次の項目を出力します。

```
1884 \newcommand{\l@figure}{\@dottedtocline{1}{1\zw}{3.683\zw}}
```

`\listoftables` 表目次を出力します。

```
1885 \newcommand{\listoftables}{%
1886 %<*book|report>
1887 \if@twocolumn\@restonecoltrue\onecolumn
1888 \else\@restonecolfalse\fi
1889 \chapter*{\listtablename}%
1890 \@mkboth{\listtablename}{}%
1891 %</book|report>
1892 %<!*book&!report>
1893 \section*{\listtablename}%
1894 \@mkboth{\listtablename}{\listtablename}%
1895 %</!*book&!report>
1896 \@starttoc{lot}%
1897 %<book|report> \if@restonecol\twocolumn\fi
1898 }
```

`\l@table` 表目次は図目次と同じです。

```
1899 \let\l@table\l@figure
```

11.2 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。

```
1900 \newdimen\bibindent
1901 \setlength\bibindent{2\zw}
```

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

[2016-07-16] L^AT_EX 2.09 で使われていたフォントコマンドの警告を、文献スタイル (.bst) ではよく `\bf` がいまだに用いられることが多いため、`thebibliography` 環境内では例外的に出さないようにしました。

```
1902 \newenvironment{thebibliography}[1]{%
1903 \global\@jsc@warnoldfontcmdexceptiontrue
1904 \global\let\presectionname\relax
1905 \global\let\postsectionname\relax
1906 %<article|jspf> \section*{\refname}\@mkboth{\refname}{\refname}%
1907 %<*kiyou>
1908 \vspace{1.5\baselineskip}
1909 \subsubsection*{\refname}\@mkboth{\refname}{\refname}%
1910 \vspace{0.5\baselineskip}
1911 %</kiyou>
1912 %<book|report> \chapter*{\bibname}\@mkboth{\bibname}{}%
1913 %<book|report> \addcontentsline{toc}{chapter}{\bibname}%
```

```

1914 \list{\@biblabel{\@arabic\c@enumiv}}%
1915     {\settowidth\labelwidth{\@biblabel{#1}}%
1916     \leftmargin\labelwidth
1917     \advance\leftmargin\labelsep
1918     \@openbib@code
1919     \usecounter{enumiv}%
1920     \let\p@enumiv\@empty
1921     \renewcommand\theenumiv{\@arabic\c@enumiv}}%
1922 %<kiyou> \small
1923 \sloppy
1924 \clubpenalty4000
1925 \@clubpenalty\clubpenalty
1926 \widowpenalty4000%
1927 \sfcode`. \. \@m}
1928 {\def\@noitemerr
1929  {\@latex@warning{Empty `thebibliography' environment}}%
1930 \endlist
1931 \global\@jsc@warnoldfontcmdexceptionfalse}

```

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

```
1932 \newcommand{\newblock}{\hskip .11em\@plus.33em\@minus.07em}
```

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

```
1933 \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。

```
1934 % \def\@biblabel#1{\inhibitglue [#1] \inhibitglue}
```

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

```

1935 % \def\@citex[#1]#2{\leavevmode
1936 %   \let\@citea\@empty
1937 %   \@cite{\@for\@citeb:=#2\do
1938 %     {\@citea\def\@citea{ \inhibitglue\penalty\@m }%
1939 %     \edef\@citeb{\expandafter\@firstofone\@citeb\@empty}}%
1940 %     \iffiles\immediate\write\@auxout{\string\citation{\@citeb}}\fi
1941 %     \ifundefined{b@\@citeb}{\mbox{\normalfont\bfseries ?}}%
1942 %     \G@refundefinedtrue
1943 %     \@latex@warning
1944 %       {Citation `@\@citeb' on page \thepage \space undefined}}%
1945 %     {\@citeofmt{\csname b@\@citeb\endcsname}}}{#1}}
1946 % \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。

```
1947 % \DeclareRobustCommand\cite{\unskip
1948 %   \@ifnextchar [{\@tempwatrue\@citex}{\@tempwafalse\@citex[]}]
1949 % \def\@cite#1#2{${\hbox{\scriptsize{#1}\if@tempwa
1950 %   , \inhibitglue\ #2\fi) }}$}
```

11.3 索引

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

```
1951 \newenvironment{theindex}{% 索引を 3 段組で出力する環境
1952   \if@twocolumn
1953     \onecolumn\@restonecolfalse
1954   \else
1955     \clearpage\@restonecoltrue
1956   \fi
1957   \columnseprule.4pt \columnsep 2\zw
1958   \ifx\multicols\@undefined
1959 %<book|report>   \twocolumn[\@makeschapterhead{\indexname}]%
1960 %<book|report>   \addcontentsline{toc}{chapter}{\indexname}]%
1961 %<!book&!report> \def\presectionname{}\def\postsectionname{}%
1962 %<!book&!report> \twocolumn[\section*{\indexname}]%
1963   \else
1964     \ifdim\textwidth<\fullwidth
1965       \setlength{\evensidemargin}{\oddsidemargin}
1966       \setlength{\textwidth}{\fullwidth}
1967       \setlength{\linewidth}{\fullwidth}
1968 %<book|report>   \begin{multicols}{3}[\chapter*{\indexname}]%
1969 %<book|report>   \addcontentsline{toc}{chapter}{\indexname}]%
1970 %<!book&!report> \def\presectionname{}\def\postsectionname{}%
1971 %<!book&!report> \begin{multicols}{3}[\section*{\indexname}]%
1972   \else
1973 %<book|report>   \begin{multicols}{2}[\chapter*{\indexname}]%
1974 %<book|report>   \addcontentsline{toc}{chapter}{\indexname}]%
1975 %<!book&!report> \def\presectionname{}\def\postsectionname{}%
1976 %<!book&!report> \begin{multicols}{2}[\section*{\indexname}]%
1977   \fi
1978   \fi
1979 %<book|report>   \@mkboth{\indexname}{}%
1980 %<!book&!report> \@mkboth{\indexname}{\indexname}%
1981   \plainifnotempty % \thispagestyle{plain}
1982   \parindent\z@
1983   \parskip\z@ \@plus .3\jsc@mpt\relax
1984   \let\item\@idxitem
1985   \raggedright
1986   \footnotesize\narrowbaselines
```

```

1987 }{
1988   \ifx\multicols\@undefined
1989     \if@restonecol\onecolumn\fi
1990   \else
1991     \end{multicols}
1992   \fi
1993   \clearpage
1994 }

```

`\@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 1995 \newcommand{\@idxitem}{\par\hangindent 4\zw} % 元 40pt
\subsubitem 1996 \newcommand{\subitem}{\@idxitem \hspace*{2\zw}} % 元 20pt
1997 \newcommand{\subsubitem}{\@idxitem \hspace*{3\zw}} % 元 30pt

```

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

```
1998 \newcommand{\indexspace}{\par \vskip 10\jsc@empt \@plus5\jsc@empt \@minus3\jsc@empt\relax}
```

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

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

```

1999 \newcommand\seename{\if@english see\else →\fi}
2000 \newcommand\alsoname{\if@english see also\else →\fi}

```

11.4 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。

```

2001 \let\footnotes@ve=\footnote
2002 \def\footnote{\inhibitglue\footnotes@ve}
2003 \let\footnotemarks@ve=\footnotemark
2004 \def\footnotemark{\inhibitglue\footnotemarks@ve}

```

`\@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 pTeX 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)。

[2014-07-02 LTJ] `\ifydir` を使わない形に書換えました。

[2016-07-11] コミュニティ版 pLaTeX の変更に従いました (Thanks: 角藤さん)。

[2016-08-27 LTJ] 結果的に `\@makefnmark` の定義が LuaTeX-japan 本体 (lltjcore.sty) 中のものと全く同じになっていたため、削除します。

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

```
2005 \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。

```
2006 % \def\thefootnote{\ifnum\c@footnote>\z@ 注\kern0.1\zw\@arabic\c@footnote\fi}
```

`\footnoterule` 本文と脚注の間の罫線です。

```
2007 \renewcommand{\footnoterule}{%
2008   \kern-3\jsc@empt
2009   \hrule width .4\columnwidth height 0.4\jsc@empt
2010   \kern 2.6\jsc@empt}
```

`\c@footnote` 脚注番号は章ごとにリセットされます。

[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。

```
2011 %<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)

[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、追従します。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。

```
2012 \long\def\@footnotetext{%
2013   \insert\footins\bgroup
2014     \normalfont\footnotesize
2015     \interlinepenalty\interfootnotelinepenalty
2016     \splittopskip\footnotesep
2017     \splitmaxdepth \dp\strutbox \floatingpenalty \@MM
2018     \hsize\columnwidth \parboxrestore
2019     \def\@currentcounter{footnote}%
2020     \protected@edef\@currentlabel{%
2021       \csname p@footnote\endcsname\@thefnmark
2022     }%
2023     \color@begingroup
```



```

2024 \makefnfoot{%
2025 \rule{\z@\footnotesep\ignorespaces}%
2026 \futurelet\jsc@next\jsc@foot}
2027 \def\jsc@foot{\ifcat\bgroup\noexpand\jsc@next \let\jsc@next\jsc@foot
2028 \else \let\jsc@next\jsc@foot\fi \jsc@next}
2029 \def\jsc@foot@prefix{\inhibitglue\ignorespaces}
2030 \def\jsc@foot@bgroup\aftergroup\jsc@foot\afterassignment\jsc@foot@prefix\let\jsc@next}
2031 \def\jsc@foot#1{\jsc@foot@prefix#1\jsc@foot}
2032 \def\jsc@foot{\@finalstrut\strutbox\color@endgroup\egroup}

```

`\makefnfoot` 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。

```

2033 \newcommand\makefnfoot[1]{%
2034 \advance\leftskip 3\zw
2035 \parindent 1\zw
2036 \noindent
2037 \hb@xt@\z@\{ \hss\makefnmark\hskip0.3\zw}#1}

```

`\xfootnotetext` 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。

```

2038 % \def\xfootnotetext[#1]{%
2039 % \begingroup
2040 % \ifnum#1>\z@
2041 % \csname c@\mpfn\endcsname #1\relax
2042 % \unrestored@protected@xdef\thefnmark{\thempfn}%
2043 % \else
2044 % \unrestored@protected@xdef\thefnmark{}%
2045 % \fi
2046 % \endgroup
2047 % \footnotetext}

```

12 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。

[2012-04-24 LTJ] Lua_T_EX-j_a では JFM に段落開始時の括弧類の字下げ幅をコントロールする機能がありますが、`\item` 直後ではラベル用のボックスが段落先頭になるため、うまく働きませんでした。形を変えて復活させます。

[2017-04-03 LTJ] 従来クラスファイルで定義していた `\inhibitglue` は、Lua_T_EX-j_a のコアに `\ltjfakeparbegin` として正式に追加されたのでリネームします。

`\item` 命令の直後です。

```

2048 \let\inhibitglue=\ltjfakeparbegin

```

```

2049 \def\@item[#1]{%
2050   \if@noperitem
2051     \@donoperitem
2052   \else
2053     \if@inlabel
2054       \indent \par
2055     \fi
2056     \ifhmode
2057       \unskip\unskip \par
2058     \fi
2059     \if@newlist
2060       \if@nobreak
2061         \@nbitem
2062       \else
2063         \addpenalty\@beginparpenalty
2064         \addvspace\@topsep
2065         \addvspace{-\parskip}%
2066       \fi
2067     \else
2068       \addpenalty\@itempenalty
2069       \addvspace\itemsep
2070     \fi
2071     \global\@inlabeltrue
2072   \fi
2073   \everypar{%
2074     \@minipagefalse
2075     \global\@newlistfalse
2076     \if@inlabel
2077       \global\@inlabelfalse
2078       {\setbox\z@\lastbox
2079        \ifvoid\z@
2080          \kern-\itemindent
2081        \fi}%
2082     \box\@labels
2083     \penalty\z@
2084   \fi
2085   \if@nobreak
2086     \@nobreakfalse
2087     \clubpenalty \@M
2088   \else
2089     \clubpenalty \@clubpenalty
2090     \everypar{]%
2091   \fi\ltjfakeparbegin}%
2092 \if@noitemarg
2093   \@noitemargfalse
2094   \if@nmbbrlist
2095     \refstepcounter\@listctr
2096   \fi
2097 \fi

```

```

2098 \sbox\@tempboxa{\makelabel{#1}}%
2099 \global\setbox\@labels\hbox{%
2100   \unhbox\@labels
2101   \hskip \itemindent
2102   \hskip -\labelwidth
2103   \hskip -\labelsep
2104   \ifdim \wd\@tempboxa >\labelwidth
2105     \box\@tempboxa
2106   \else
2107     \hbox to\labelwidth {\unhbox\@tempboxa}%
2108   \fi
2109   \hskip \labelsep}%
2110 \ignorespaces}

```

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

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

[2016-12-05 LTJ] 本家 [2016-11-29], `lltjcore.sty` での変更に従わせてます。

[2017-02-18 LTJ] `lltjcore.sty` 側に戻したのを忘れていました。

```

2111 \def\@gnewline #1{%
2112   \ifvmode
2113     \@nolnerr
2114   \else
2115     \unskip \reserved@e {\reserved@f#1}\nobreak \hfil \break \null
2116     \inhibitglue \ignorespaces
2117   \fi}

```

13 いろいろなロゴ

\LaTeX 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。

[2016-07-21 LTJ] `jsclasses` と `Lua \TeX -ja` の更新タイミングが一致しない可能性を考慮し、`jslogo` パッケージが存在しない場合は旧来の定義をそのまま使うことにしました。

```

2118 \IfFileExists{jslogo.sty}{\@jslogofalse}%
2119 \if@jslogo
2120   \RequirePackage{jslogo}
2121   \def\小{\jslg@small}
2122   \def\上小{\jslg@uppersmall}
2123 \else

```

以下は `jslogo` パッケージがない場合の定義です。

\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です。

```
\上小 2124 \def\小#1{\hbox{${\m@th$%
2125 \csname S@f@size\endcsname
2126 \fontsize\sf@size\z@
2127 \math@fontsfalse\selectfont
2128 #1}}
2129 \def\上小#1{\sbox\z@ T\vbox to\ht0{\小{#1}\vss}}
```

\TeX これらは ltlogos.dtx で定義されていますが、Times や Helvetica でも見栄えがするよう
\LaTeX に若干変更しました。

[2003-06-12] Palatino も加えました (要調整)。

```
2130 \def\cmrTeX{%
2131 \ifdim \fontdimen\@ne\font >\z@
2132 T\kern-.25em\lower.5ex\hbox{E}\kern-.125emX\@
2133 \else
2134 T\kern-.1667em\lower.5ex\hbox{E}\kern-.125emX\@
2135 \fi}
2136 \def\cmrLaTeX{%
2137 \ifdim \fontdimen\@ne\font >\z@
2138 L\kern-.32em\上小{A}\kern-.22em\cmrTeX
2139 \else
2140 L\kern-.36em\上小{A}\kern-.15em\cmrTeX
2141 \fi}
2142 \def\sfTeX{T\kern-.1em\lower.4ex\hbox{E}\kern-.07emX\@}
2143 \def\sfLaTeX{L\kern-.25em\上小{A}\kern-.08em\sfTeX}
2144 \def\ptmTeX{%
2145 \ifdim \fontdimen\@ne\font >\z@
2146 T\kern-.12em\lower.37ex\hbox{E}\kern-.02emX\@
2147 \else
2148 T\kern-.07em\lower.37ex\hbox{E}\kern-.05emX\@
2149 \fi}
2150 \def\ptmLaTeX{%
2151 \ifdim \fontdimen\@ne\font >\z@
2152 L\kern-.2em\上小{A}\kern-.1em\ptmTeX
2153 \else
2154 L\kern-.3em\上小{A}\kern-.1em\ptmTeX
2155 \fi}
2156 \def\pncTeX{%
2157 \ifdim \fontdimen\@ne\font >\z@
2158 T\kern-.2em\lower.5ex\hbox{E}\kern-.08emX\@
2159 \else
2160 T\kern-.13em\lower.5ex\hbox{E}\kern-.13emX\@
2161 \fi}
2162 \def\pncLaTeX{%
2163 \ifdim \fontdimen\@ne\font >\z@
2164 L\kern-.3em\上小{A}\kern-.1em\pncTeX
2165 \else
2166 L\kern-.3em\上小{A}\kern-.1em\pncTeX
```

```

2167 \fi}
2168 \def\pplTeX{%
2169 \ifdim \fontdimen\@ne\font >\z@
2170 T\kern-.17em\lower.32ex\hbox{E}\kern-.15emX\@
2171 \else
2172 T\kern-.12em\lower.34ex\hbox{E}\kern-.1emX\@
2173 \fi}
2174 \def\pplLaTeX{%
2175 \ifdim \fontdimen\@ne\font >\z@
2176 L\kern-.27em\上小{A}\kern-.12em\pplTeX
2177 \else
2178 L\kern-.3em\上小{A}\kern-.15em\pplTeX
2179 \fi}
2180 \def\ugmTeX{%
2181 \ifdim \fontdimen\@ne\font >\z@
2182 T\kern-.1em\lower.32ex\hbox{E}\kern-.06emX\@
2183 \else
2184 T\kern-.12em\lower.34ex\hbox{E}\kern-.1emX\@
2185 \fi}
2186 \def\ugmLaTeX{%
2187 \ifdim \fontdimen\@ne\font >\z@
2188 L\kern-.2em\上小{A}\kern-.13em\ugmTeX
2189 \else
2190 L\kern-.3em\上小{A}\kern-.13em\ugmTeX
2191 \fi}
2192 \DeclareRobustCommand{\TeX}{%
2193 \def\@tempa{cmr}%
2194 \ifx\f@family\@tempa\cmrTeX
2195 \else
2196 \def\@tempa{ptm}%
2197 \ifx\f@family\@tempa\ptmTeX
2198 \else
2199 \def\@tempa{txr}%
2200 \ifx\f@family\@tempa\ptmTeX
2201 \else
2202 \def\@tempa{pnc}%
2203 \ifx\f@family\@tempa\pncTeX
2204 \else
2205 \def\@tempa{ppl}%
2206 \ifx\f@family\@tempa\pplTeX
2207 \else
2208 \def\@tempa{ugm}%
2209 \ifx\f@family\@tempa\ugmTeX
2210 \else\sfTeX
2211 \fi
2212 \fi
2213 \fi
2214 \fi
2215 \fi

```

```

2216 \fi}
2217
2218 \DeclareRobustCommand{\LaTeX}{%
2219 \def\@tempa{cmr}%
2220 \ifx\f@family\@tempa\cmrLaTeX
2221 \else
2222 \def\@tempa{ptm}%
2223 \ifx\f@family\@tempa\ptmLaTeX
2224 \else
2225 \def\@tempa{txr}%
2226 \ifx\f@family\@tempa\ptmLaTeX
2227 \else
2228 \def\@tempa{pnc}%
2229 \ifx\f@family\@tempa\pncLaTeX
2230 \else
2231 \def\@tempa{ppl}%
2232 \ifx\f@family\@tempa\pplLaTeX
2233 \else
2234 \def\@tempa{ugm}%
2235 \ifx\f@family\@tempa\ugmLaTeX
2236 \else\sfLaTeX
2237 \fi
2238 \fi
2239 \fi
2240 \fi
2241 \fi
2242 \fi}

```

\LaTeXe \LaTeXe コマンドの `\mbox{\m@th ...}` で始まる新しい定義では直後の和文との間に `xkanjiskip` が入りません。また、`mathptmx` パッケージなどと併用すると、最後の ε が下がりがすぎてしまいます。そのため、ちょっと手を加えました。

```

2243 \DeclareRobustCommand{\LaTeXe}{\mbox{%
2244 \if b\expandafter\@car\f@series\@nil\boldmath\fi
2245 \LaTeX\kern.15em2\raisebox{-.37ex}{\textstyle\varepsilon}}}%

```

\pTeX pTeX, pLATEX 2_ε のロゴを出す命令です。

```

\pLaTeX 2246 \def\pTeX{p\kern-.05em\TeX}
\pLaTeXe 2247 \def\pLaTeX{p\LaTeX}
2248 \def\pLaTeXe{p\LaTeXe}

```

\AmSTeX `amstex.sty` で定義されています。

```

2249 \def\AmSTeX{\protect\AmS-\protect\TeX{}}

```

\BibTeX これらは `doc.dtx` から取ったものです。ただし、\BibTeX だけはちょっと修正しました。

```

\SliTeX 2250 % \@ifundefined{BibTeX}
2251 % {\def\BibTeX{\rmfamily B\kern-.05em%
2252 % \textsc{i\kern-.025em b}\kern-.08em%
2253 % T\kern-.1667em\lower.7ex\hbox{E}\kern-.125emX}}{}
2254 \DeclareRobustCommand{\BibTeX}{B\kern-.05em\小{I\kern-.025em B}%

```

```

2255 \ifx\f@family\cmr\kern-.08em\else\kern-.15em\fi\TeX}
2256 \DeclareRobustCommand{\SliTeX}{%
2257 S\kern-.06emL\kern-.18em\上小{I}\kern-.03em\TeX}

```

jslogo パッケージがない場合の定義はここで終わりです。

```
2258 \fi
```

14 初期設定

■いろいろな語

```

\prepartname
\postpartname 2259 \newcommand{\prepartname}{\if@english Part~\else 第 \fi}
\prechaptername 2260 \newcommand{\postpartname}{\if@english\else 部 \fi}
2261 %<book|report>\newcommand{\prechaptername}{\if@english Chapter~\else 第 \fi}
\postchaptername 2262 %<book|report>\newcommand{\postchaptername}{\if@english\else 章 \fi}
\presectionname 2263 \newcommand{\presectionname}{}% 第
\postsectionname 2264 \newcommand{\postsectionname}{}% 節

\contentsname
\listfigurename 2265 \newcommand{\contentsname}{\if@english Contents\else 目次 \fi}
\listtablename 2266 \newcommand{\listfigurename}{\if@english List of Figures\else 図目次 \fi}
2267 \newcommand{\listtablename}{\if@english List of Tables\else 表目次 \fi}

\refname
\bibName 2268 \newcommand{\refname}{\if@english References\else 参考文献 \fi}
\indexname 2269 \newcommand{\bibname}{\if@english Bibliography\else 参考文献 \fi}
2270 \newcommand{\indexname}{\if@english Index\else 索引 \fi}

\figurename
\tablename 2271 %<!jspf>\newcommand{\figurename}{\if@english Fig.~\else 図 \fi}
2272 %<jspf>\newcommand{\figurename}{Fig.~}
2273 %<!jspf>\newcommand{\tablename}{\if@english Table~\else 表 \fi}
2274 %<jspf>\newcommand{\tablename}{Table~}

\appendixname
\abstractname 2275 % \newcommand{\appendixname}{\if@english Appendix~\else 付録 \fi}
2276 \newcommand{\appendixname}{\if@english \else 付録 \fi}
2277 %<!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 5 月 6 日で、和暦では令和 7 年 5 月 6 日です。

```

\today
2278 \newif\if 西暦 \西暦 true
2279 \def\西暦{\西暦 true}

```

```

2280 \def\和暦{\西暦 false}
2281 \newcount\heisei \heisei\year \advance\heisei-1988\relax
2282 \def\pltx@today@year@#1{%
2283   \ifnum\numexpr\year-#1=1 元 \else
2284     \ifnum\ltjgetparameter{direction}=3
2285       \kansuji\numexpr\year-#1\relax
2286     \else
2287       \number\numexpr\year-#1\relax\nobreak
2288     \fi
2289   \fi 年
2290 }
2291 \def\pltx@today@year{%
2292   \ifnum\numexpr\year*10000+\month*100+\day<19890108
2293     昭和 \pltx@today@year@{1925}%
2294   \else\ifnum\numexpr\year*10000+\month*100+\day<20190501
2295     平成 \pltx@today@year@{1988}%
2296   \else
2297     令和 \pltx@today@year@{2018}%
2298   \fi\fi}
2299 \def\today{%
2300   \if@english
2301     \ifcase\month\or
2302       January\or February\or March\or April\or May\or June\or
2303       July\or August\or September\or October\or November\or December\fi
2304     \space\number\day, \number\year
2305   \else\if 西暦
2306     \ifnum\ltjgetparameter{direction}=3 \kansuji\year
2307     \else\number\year\nobreak\fi 年
2308   \else
2309     \pltx@today@year
2310   \fi
2311   \ifnum\ltjgetparameter{direction}=3
2312     \kansuji\month 月
2313     \kansuji\day 日
2314   \else
2315     \number\month\nobreak 月
2316     \number\day\nobreak 日
2317   \fi\fi}

```

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

```

2318 \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。p \LaTeX の標準時と同じようにボトムフロートの下
に脚注が組まれるようにします。L \TeX 2 ϵ 2025-06-01 より以前のバージョンでは stfloats
パッケージを使います。

[2017-02-19] pL^AT_EX と LuaT_EX-ja の \makecol が違うことを考慮していませんでした。

[2025-03-28 LTJ] L^AT_EX 2_ε 2025-06-01 での変更に従。

```
2319 %<article>\if@slide \pagestyle{empty} \else \pagestyle{plain} \fi
2320 %<book>\if@report \pagestyle{plain} \else \pagestyle{headings} \fi
2321 %<report|kiyou>\pagestyle{plain}
2322 %<jspf>\pagestyle{headings}
2323 \pagenumbering{arabic}
2324 \@ifl@t@r\fmtversion{2025-06-01}{%
2325   \AssignSocketPlug{build/column/outputbox}{floats-footnotes-plaintext}
2326 }{%
2327   \fnfixbottomtrue % 2017-02-19
2328   \IfFileExists{stfloats.sty}{\RequirePackage{stfloats}\fnbelowfloat}{}
2329 }
2330 \if@twocolumn
2331   \twocolumn
2332   \sloppy
2333   \flushbottom
2334 \else
2335   \onecolumn
2336   \raggedbottom
2337 \fi
2338 \if@slide
2339   \renewcommand\kanjifamilydefault{\gtdefault}
2340   \renewcommand\familydefault{\sfdefault}
2341   \raggedright
2342   \ltj@setpar@global
2343   \ltjsetxkanjiskip0.1em\relax
2344 \fi
```

以上です。