

Lua \LaTeX -ja 用 jsclasses 互換クラス

Lua \TeX -ja プロジェクト

2019/08/12

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	50
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	62
11.1	目次の類	62
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	78

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 $\text{Lua}\TeX$ -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` ($\text{Lua}\TeX$ -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 $\text{Lua}\TeX$ -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。
- $\text{Lua}\TeX$ -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 へパッチを当てます。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 `bxjcls` 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-já の読み込み

和文スケール値を設定した後に、LuaTeX-já 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 1m^2 、縦横比 $1:\sqrt{2}$ の長方形の辺の長さを mm 単位に切り捨てたものです。これを基準として順に半載しては mm 単位に切り捨てたものが A1, A2, …です。

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

デフォルトは `a4paper` です。

b5var (B5 変形, 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}}

```

■**オプ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 以外の場合にオプ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] `var twoside` 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{var twoside}{\@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、`lATEX` classes 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` 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\\\@eqnrcr
136   $$\everycr{\halign to\displaywidth\bgroup
137     \hskip\@centering$\displaystyle\tabskip\z@skip{##}$\@eqnrel
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\belowdisplaysshortskip{\abovedisplayskip}%
160     \setlength\abovedisplaysshortskip{\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、LaTeX 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}{\ptexjistruer}
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 t = lua.get_functions_table()
250   t[\the\ltjs@@magnify@font@calc] = function()
251     tex.sprint(-2,math.floor(0.5+mpt*getdimen('dimen')))
252   end
253   function luatexja.ltjs_unmagnify_fsize(a)
254     local s = luatexja.print_scaled(math.floor(0.5+a/mpt*65536))
255     tex.sprint(-2, (s:match('%.0$')) and s:sub(1,-3) or s )
256   end
257 }
258 \endgroup
259 \def\ltjs@magnify@external@font#1 at#2 at#3@nil{%
260   \def\@tempa{#1}\def\@tempb{#2}%
261   \ifx\@tempb\@empty
262     \edef\@tempb{ scaled\directlua{%
263       tex.sprint(-2,math.floor(0.5+\jsc@magscale*1000))
264     }}%
265   \else
266     \dimen@\@tempb\relax
267     \edef\@tempb{ at\luafunction\ltjs@@magnify@font@calc sp}%
268   \fi
269   \edef\@tempa{\def\noexpand\external@font{\@tempa\@tempb}}%
270 }
271 \let\ltjs@orig@get@external@font=\get@external@font
272 \def\get@external@font{%
273   \edef\f@size{\directlua{luatexja.ltjs_unmagnify_fsize(\f@size)}}%
274   \ltjs@orig@get@external@font
275   \begingroup
276     \edef\@tempa{\external@font\space at\space at}%
277     \expandafter\ltjs@magnify@external@font\@tempa@nil
278   \expandafter\endgroup\@tempa
279 }
```

280 \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 281 \def\jsc@smallskip{\vspace\jsc@smallskipamount}
\jsc@bigskip 282 %\def\jsc@medskip{\vspace\jsc@medskipamount}
                283 %\def\jsc@bigskip{\vspace\jsc@bigskipamount}

\jsc@smallskipamount
\jsc@medskipamount 284 \newskip\jsc@smallskipamount
\jsc@bigskipamount 285 \jsc@smallskipamount=3\jsc@empt plus 1\jsc@empt minus 1\jsc@empt
                286 %\newskip\jsc@medskipamount
                287 %\jsc@medskipamount =6\jsc@empt plus 2\jsc@empt minus 2\jsc@empt
                288 %\newskip\jsc@bigskipamount
                289 %\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 インチ足しておきます。
`\pageheight`
`\stockwidth` [2015-10-18 LTJ] LuaTeX 0.81.0 ではプリミティブの名称変更がされたので、それに合わせておきます。
`\stockheight`

[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。

```
290 \iftombow
291   \newlength{\stockwidth}
292   \newlength{\stockheight}
293   \setlength{\stockwidth}{\paperwidth}
294   \setlength{\stockheight}{\paperheight}
295   \advance \stockwidth 2in
296   \advance \stockheight 2in
297   \ifdefined\pdfpagewidth
298     \setlength{\pdfpagewidth}{\stockwidth}
299     \setlength{\pdfpageheight}{\stockheight}
300   \else
301     \setlength{\pagewidth}{\stockwidth}
302     \setlength{\pageheight}{\stockheight}
303   \fi
304 \else
305   \ifdefined\pdfpagewidth
306     \setlength{\pdfpagewidth}{\paperwidth}
307     \setlength{\pdfpageheight}{\paperheight}
```

```

308 \else
309   \setlength{\pagewidth}{\paperwidth}
310   \setlength{\pageheight}{\paperheight}
311 \fi
312 \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。

p $\text{T}_{\text{E}}\text{X}$ (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, Lua $\text{T}_{\text{E}}\text{X}$ -ja 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@stdgfont` 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なく, 何らかの理由で非埋め込みフォントが正しく利用できない場合にのみ `luatexja.cfg` によってセットされるものです。

[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。

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 Lua $\text{T}_{\text{E}}\text{X}$ -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。

```
313 \expandafter\let\csname JY3/mc/m/n/10\endcsname\relax
314 \ifmingoth
315   \DeclareFontShape{JY3}{mc}{m}{n}{<-> s * [\Cjascale] \ltj@stdmcfont:jfm=min}{}
316   \DeclareFontShape{JY3}{gt}{m}{n}{<-> s * [\Cjascale] \ltj@stdgtfont:jfm=min}{}
317 \else
318   \ifptexjis
319     \DeclareFontShape{JY3}{mc}{m}{n}{<-> s * [\Cjascale] \ltj@stdmcfont:jfm=jis}{}
320     \DeclareFontShape{JY3}{gt}{m}{n}{<-> s * [\Cjascale] \ltj@stdgtfont:jfm=jis}{}
321   \fi
322 \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を `\jttddefault` とし, 通常のゴシック体と別にできるようにしました。`\jttddefault` 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になってしまうという弊害がありました。`amsthm` を使わない場合は定理の本文が明朝になるように `\newtheorem` 環境を手直ししてしのいでいましたが, $\text{T}_{\text{E}}\text{X}$ が数学で多用されることを考えると, イタリック体に明朝体を当てたほうが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。

```
323 \DeclareFontShape{JY3}{mc}{bx}{n}{<->ssub*gt/m/n}{}
324 \DeclareFontShape{JY3}{gt}{bx}{n}{<->ssub*gt/m/n}{}
325 \DeclareFontShape{JY3}{mc}{b}{n}{<->ssub*gt/m/n}{}
326 \DeclareFontShape{JY3}{mc}{b}{n}{<->ssub*gt/m/n}{}
327 \DeclareFontShape{JY3}{mc}{m}{it}{<->ssub*mc/m/n}{}
328 \DeclareFontShape{JY3}{mc}{m}{sl}{<->ssub*mc/m/n}{}
329 \DeclareFontShape{JY3}{mc}{m}{sc}{<->ssub*mc/m/n}{}
330 \DeclareFontShape{JY3}{gt}{m}{it}{<->ssub*gt/m/n}{}
331 \DeclareFontShape{JY3}{gt}{m}{sl}{<->ssub*gt/m/n}{}
332 \DeclareFontShape{JY3}{mc}{bx}{it}{<->ssub*gt/m/n}{}
333 \DeclareFontShape{JY3}{mc}{bx}{sl}{<->ssub*gt/m/n}{}
334 \DeclareFontShape{JY3}{mc}{b}{it}{<->ssub*gt/m/n}{}
335 \DeclareFontShape{JY3}{mc}{b}{sl}{<->ssub*gt/m/n}{}
336 \DeclareFontShape{JT3}{mc}{bx}{n}{<->ssub*gt/m/n}{}
337 \DeclareFontShape{JT3}{gt}{bx}{n}{<->ssub*gt/m/n}{}
338 \DeclareFontShape{JT3}{mc}{b}{n}{<->ssub*gt/m/n}{}

```



```

339 \DeclareFontShape{JT3}{mc}{b}{n}{<->ssub*gt/m/n}{}
340 \DeclareFontShape{JT3}{mc}{m}{it}{<->ssub*mc/m/n}{}
341 \DeclareFontShape{JT3}{mc}{m}{sl}{<->ssub*mc/m/n}{}
342 \DeclareFontShape{JT3}{mc}{m}{sc}{<->ssub*mc/m/n}{}
343 \DeclareFontShape{JT3}{gt}{m}{it}{<->ssub*gt/m/n}{}
344 \DeclareFontShape{JT3}{gt}{m}{sl}{<->ssub*gt/m/n}{}
345 \DeclareFontShape{JT3}{mc}{bx}{it}{<->ssub*gt/m/n}{}
346 \DeclareFontShape{JT3}{mc}{bx}{sl}{<->ssub*gt/m/n}{}
347 \DeclareFontShape{JT3}{mc}{b}{it}{<->ssub*gt/m/n}{}
348 \DeclareFontShape{JT3}{mc}{b}{sl}{<->ssub*gt/m/n}{}
349 \renewcommand\jttdefault{\gtdefault}\@ltj@match@familytrue

```

LuaTeX-ja 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、jscsses 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。

jscsses.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-ja では削除しています。

```

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

```

`\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。

```

356 % \DeclareTextCommand{\textsterling}{OT1}{\itshape\char`\$}

```

アスキーの `kinsoku.dtx` では「」「“」「”」前後のペナルティが 5000 になっていたのですが、`jscsses.dtx` ではそれを 10000 に補正していました。しかし、LuaTeX-ja 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。

```

357 \ltjsetparameter{jaxspmode={`!,2}}
358 \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。

```

359 \ltjsetparameter{alxspmode={`+,3}}
360 \ltjsetparameter{alxspmode={`%,3}}

```

jsclasses.dtx では 80~ff の文字の `\xspace` 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, L^AT_EX 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 L^AT_EX 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。

```
361 \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 L^AT_EX の内部命令 `\@xpt` を使っています。この `\@xpt` の類は次のものがあり, L^AT_EX 本体で定義されています。

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

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

`kanjiskip` は `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の, 追い出しと追い込みの混在が生じ, 統一性を欠きます。なるべく追い出しになるようにプラスの値だけにしたいところですが, ごくわずかなマイナスは許すことにしました。

`xkanjiskip` については, 四分つまり全角の 1/4 を標準として, 追い出すために三分あるいは二分まで延ばすのが一般的ですが, ここでは Times や Palatino のスペースがほぼ四分であることに着目して, これに一致させています。これなら書くときにスペースを空けても空けなくても同じ出力になります。

`\parindent` については, 0 (以下) でなければ全角幅 (`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。

```
362 \def\@setfontsize#1#2#3{%
363 % \@nomath#1%
364 \ifx\protect\@typesetprotect
365   \let\@currsize#1%
366 \fi
367 \fontsize{#2}{#3}\selectfont
368 \ifdim\parindent>\z@
369   \if@english
370     \parindent=1em
371   \else
372     \parindent=1\zw
373 \fi
374 \fi
375 \ltj@setpar@global
376 \ltjsetkanjiskip\z@ plus .1\zw minus .01\zw
377 \@tempskipa=\ltjgetparameter{xkanjiskip}
378 \ifdim\@tempskipa>\z@
379   \if@slide
380     \ltjsetxkanjiskip .1em
381   \else
382     \ltjsetxkanjiskip .25em plus .15em minus .06em
383 \fi
384 \fi}
```

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

```
385 \def\jsc@setfontsize#1#2#3{%
386   \@setfontsize#1{#2\jsc@mpt}{#3\jsc@mpt}}
```

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

```
387 \emergencystretch 3\zw
```

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

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

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

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

[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。

```
388 \newif\ifnarrowbaselines
```

```

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

```

`\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。

```

409 \renewcommand{\normalsize}{%
410 \ltj@ifnarrowbaselines
411 {\jsc@setfontsize\normalsize\@xpt\@xiipt}%
412 {\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。

```

413 \abovedisplayskip 11\jsc@mpt \@plus3\jsc@mpt \@minus4\jsc@mpt
414 \abovedisplayshortskip \z@ \@plus3\jsc@mpt
415 \belowdisplayskip 9\jsc@mpt \@plus3\jsc@mpt \@minus4\jsc@mpt
416 \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。

```

417 \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。

```
418 \mcfamily\selectfont\normalsize
```

`\Cht` 基準となる長さの設定をします。lltjfont.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 419 \setbox0\hbox{漢}
420 \setlength\Cht{\ht0}
421 \setlength\Cdp{\dp0}
422 \setlength\Cwd{\wd0}
423 \setlength\Cvs{\baselineskip}
424 \setlength\Chs{\wd0}
425 \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。

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

```
440 \newcommand{\footnotesize}{%
441   \ltj@ifnarrowbaselines
442   %<!kiyou>   {\jsc@setfontsize\footnotesize\@viiipt{9.5}}%
443   %<kiyou>    {\jsc@setfontsize\footnotesize{8.8888}{11}}%
444   %<!kiyou>   {\jsc@setfontsize\footnotesize\@viiipt{11}}%
445   %<kiyou>    {\jsc@setfontsize\footnotesize{8.8888}{13.2418}}%
446   \abovedisplayskip 6\jsc@empt \@plus2\jsc@empt \@minus3\jsc@empt
447   \abovedisplayshortskip \z@ \@plus2\jsc@empt
448   \belowdisplayskip \abovedisplayskip
449   \belowdisplayshortskip \belowdisplayskip
```

```

450 \def\@listi{\leftmargin\leftmargini
451         \topsep \z@
452         \parsep \z@
453         \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。

```

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

```

469 \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
あればお教えください。

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

```

470 % \newcommand{\headfont}{\bfseries}
471 \newcommand{\headfont}{\gtfamily\sffamily}
472 % \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。`\columnseprule` の幅の罫線が引かれます。

```
473 %<!kiyou>\setlength\columnsep{2\zw}
474 %<kiyou>\setlength\columnsep{28truebp}
475 \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。

```
\lineskip 476 \setlength\lineskip{1\jsc@mp}
\normallineskip 477 \setlength\normallineskip{1\jsc@mp}
\normallineskiplimit 478 \setlength\normallineskiplimit{1\jsc@mp}
479 \setlength\normallineskiplimit{1\jsc@mp}
```

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

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

とすると、行送りが通常の 2 倍になります。ただし、これを設定すると、たとえば `\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。

```
480 \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、ここではゼロにしました。`\parindent` は段落の先頭の字下げ幅です。

```
481 \setlength\parskip{\z@}
482 \if@slide
483 \setlength\parindent{0\zw}
484 \else
485 \setlength\parindent{1\zw}
486 \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ます。ここはオリジナル通りです。

```
\@medpenalty 487 \@lowpenalty 51
488 \@medpenalty 151
489 \@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。

```
490 % \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。

```
491 % \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にすると、本文中に \int のような高い文字が入ったときに 1 行目のベースラインが他のページより下がってしまいます。ここでは本文の公称フォントサイズ (10pt) にします。

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

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

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

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

```
498 %<*article|kiyou>
499 \if@slide
500   \setlength\footskip{\z@}
501 \else
502   \setlength\footskip{0.03367\paperheight}
503   \ifdim\footskip<\baselineskip
504     \setlength\footskip{\baselineskip}
505   \fi
506 \fi
507 %</article|kiyou>
508 %<jspf>\setlength\footskip{9\jsc@mmm}
509 %<*book>
510 \if@report
511   \setlength\footskip{0.03367\paperheight}
512   \ifdim\footskip<\baselineskip
513     \setlength\footskip{\baselineskip}
```



```

514 \fi
515 \else
516 \setlength\footskip{\z@}
517 \fi
518 %</book>
519 %<*report>
520 \setlength\footskip{0.03367\paperheight}
521 \ifdim\footskip<\baselineskip
522 \setlength\footskip{\baselineskip}
523 \fi
524 %</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)。

```

525 %<*article>
526 \if@slide
527 \setlength\headsep{0\jsc@mpt}
528 \addtolength\headsep{-\topskip}%% added (2016-10-08)
529 \addtolength\headsep{10\jsc@mpt}%% added (2016-10-08)
530 \else
531 \setlength\headsep{\footskip}
532 \addtolength\headsep{-\topskip}
533 \fi
534 %</article>
535 %<*book>
536 \if@report
537 \setlength\headsep{\footskip}
538 \addtolength\headsep{-\topskip}
539 \else
540 \setlength\headsep{6\jsc@mmm}
541 \addtolength\headsep{-\topskip}%% added (2016-10-08)
542 \addtolength\headsep{10\jsc@mpt}%% added (2016-10-08)
543 \fi
544 %</book>
545 %<*report>
546 \setlength\headsep{\footskip}
547 \addtolength\headsep{-\topskip}
548 %</report>
549 %<*jspf>
550 \setlength\headsep{9\jsc@mmm}
551 \addtolength\headsep{-\topskip}
552 %</jspf>
553 %<*kiyou>
554 \setlength\headheight{0\jsc@mpt}
555 \setlength\headsep{0\jsc@mpt}

```

```

556 \addtolength\headsep{-\topskip}%% added (2016-10-08)
557 \addtolength\headsep{10\jsc@empt}%% added (2016-10-08)
558 %</kiyou>

```

`\maxdepth` `\maxdepth` は本文最下行の最大の深さで、plain TeX や L^AT_EX 2.09 では 4pt に固定でした。L^AT_EX 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。

```

559 \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。

```

560 \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。

```

561 %<*article>
562 \if@slide
563   \setlength\fullwidth{0.9\paperwidth}
564 \else
565   \setlength\fullwidth{0.76\paperwidth}
566 \fi
567 \if@twocolumn \@tempdima=2\zw \else \@tempdima=1\zw \fi
568 \divide\fullwidth\@tempdima \multiply\fullwidth\@tempdima
569 \setlength\textwidth{\fullwidth}
570 %</article>
571 %<*book>
572 \if@report
573   \setlength\fullwidth{0.76\paperwidth}
574 \else
575   \setlength\fullwidth{\paperwidth}
576   \addtolength\fullwidth{-36\jsc@mmm}
577 \fi
578 \if@twocolumn \@tempdima=2\zw \else \@tempdima=1\zw \fi
579 \divide\fullwidth\@tempdima \multiply\fullwidth\@tempdima
580 \setlength\textwidth{\fullwidth}
581 \if@report \else

```

```

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

```

605 %<*article|book|report>
606 \if@slide
607   \setlength{\textheight}{0.95\paperheight}
608 \else
609   \setlength{\textheight}{0.83\paperheight}
610 \fi
611 \addtolength{\textheight}{-10\jsc@empt}%% from -\topskip (2016-10-08); from -
    \headheight (2003-06-26)

```

```

612 \addtolength{\textheight}{-\headsep}
613 \addtolength{\textheight}{-\footskip}
614 \addtolength{\textheight}{-\topskip}
615 \divide\textheight\baselineskip
616 \multiply\textheight\baselineskip
617 %</article|book|report>
618 %<jspf>\setlength{\textheight}{51\baselineskip}
619 %<kiyou>\setlength{\textheight}{47\baselineskip}
620 \addtolength{\textheight}{\topskip}
621 \addtolength{\textheight}{0.1\jsc@empt}
622 %<jspf>\setlength{\mathindent}{10\jsc@mmm}

```

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

```

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

```

ですが、次のようにします。

```

623 \def\flushbottom{%
624   \def\@textbottom{\vskip \z@ \@plus.1\jsc@empt}%
625   \let\@texttop\relax}

```

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

```

626 \setlength\marginparsep{\columnsep}
627 \setlength\marginparpush{\baselineskip}

```

`\oddsidemargin` それぞれ奇数ページ、偶数ページの左マージンから 1 インチ引いた値です。片面印刷では
`\evensidemargin` `\oddsidemargin` が使われます。T_EX は上・左マージンに `1truein` を挿入しますが、トンボ関係のオプションが指定されると `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] LuaT_EX (pdfT_EX?) では `1truein` ではなく `1in` になるようです。

```

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

```

637 \setlength\marginparwidth{\paperwidth}

```

```

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

```

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

```

658 {\footnotesize}\global\setlength\footnotesep{\baselineskip}
659 \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。

```

660 \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。
661 `\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。
662 `\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。
663 `\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。
664 `\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。
665 `\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。
666 `\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。
667 `\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。
668 `\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。
669 `\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。
670 `\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・下部のフロートと本文との距離です。`\intextsep` は本文の途中に出力されるフロートと本文との距離です。

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

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

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

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

`\dbltextfloatsep` 674 `\setlength\dblfloatsep {12\jsc@empt \@plus 2\jsc@empt \@minus 2\jsc@empt}`
675 `\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` 676 `\setlength\@fptop{0\jsc@empt \@plus 1fil}`
677 `\setlength\@fpsep{8\jsc@empt \@plus 2fil}`
678 `\setlength\@fpbot{0\jsc@empt \@plus 1fil}`

`\@dblfpptop` 段抜きフロートについての値です。

`\@dblfpsep` 679 `\setlength\@dblfpptop{0\jsc@empt \@plus 1fil}`
`\@dblfpbot` 680 `\setlength\@dblfpsep{8\jsc@empt \@plus 2fil}`
681 `\setlength\@dblfpbot{0\jsc@empt \@plus 1fil}`

7 改ページ (日本語 T_EX 開発コミュニティ版のみ)

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

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

となっています。

```

682 %<*article|book|report>
683 \def\pltx@cleartorightpage{\clearpage\if@twoside
684   \unless\ifodd\numexpr\c@page+\ltjgetparameter{direction}\relax
685     \hbox{}\thispagestyle{empty}\newpage
686     \if@twocolumn\hbox{}\newpage\fi
687   \fi\fi}
688 \def\pltx@cleartoleftpage{\clearpage\if@twoside
689   \ifodd\numexpr\c@page+\ltjgetparameter{direction}\relax
690     \hbox{}\thispagestyle{empty}\newpage
691     \if@twocolumn\hbox{}\newpage\fi
692   \fi\fi}
693 \def\pltx@cleartooddpage{\clearpage\if@twoside
694   \ifodd\c@page\else
695     \hbox{}\thispagestyle{empty}\newpage
696     \if@twocolumn\hbox{}\newpage\fi
697   \fi\fi}
698 \def\pltx@cleartoevenpage{\clearpage\if@twoside
699   \ifodd\c@page
700     \hbox{}\thispagestyle{empty}\newpage
701     \if@twocolumn\hbox{}\newpage\fi

```

```
702 \fi\fi}
703 %</article|book|report>
```

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

```
704 %<*book|report>
705 \if@openleft
706 \let\cleardoublepage\pltx@cleartoleftpage
707 \else\if@openright
708 \let\cleardoublepage\pltx@cleartorightpage
709 \fi\fi
710 %</book|report>
```

8 ページスタイル

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

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

ページスタイルは `\ps@...` の形のマクロで定義されています。

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

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

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

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

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

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

```
711 % \def\ps@empty{%
712 %   \let\@mkboth\@gobbletwo
713 %   \let\@oddhead\@empty
```



```

714 % \let\@oddfoot\@empty
715 % \let\@evenhead\@empty
716 % \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。
717 \def\ps@plainfoot{%
718 \let\@mkboth\@gobbletwo
719 \let\@oddhead\@empty
720 \def\@oddfoot{\normalfont\hfil\thepage\hfil}%
721 \let\@evenhead\@empty
722 \let\@evenfoot\@oddfoot}
723 \def\ps@plainhead{%
724 \let\@mkboth\@gobbletwo
725 \let\@oddfoot\@empty
726 \let\@evenfoot\@empty
727 \def\@evenhead{%
728 \if@mparswitch \hss \fi
729 \hbox to \fullwidth{\textbf{\thepage}\hfil}%
730 \if@mparswitch\else \hss \fi}%
731 \def\@oddhead{%
732 \hbox to \fullwidth{\hfil\textbf{\thepage}}\hss}}
733 %<book>\if@report \let\ps@plain\ps@plainfoot \else \let\ps@plain\ps@plainhead \fi
734 %<!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。
735 %<*article|kiyou>
736 \if@twoside
737 \def\ps@headings{%
738 \let\@oddfoot\@empty
739 \let\@evenfoot\@empty
740 \def\@evenhead{\if@mparswitch \hss \fi
741 \underline{\hbox to \fullwidth{\textbf{\thepage}\hfil\leftmark}}}%
742 \if@mparswitch\else \hss \fi}%
743 \def\@oddhead{%
744 \underline{%
745 \hbox to \fullwidth{\rightmark}\hfil\textbf{\thepage}}\hss}%
746 \let\@mkboth\markboth
747 \def\sectionmark##1{\markboth{%
748 \ifnum \c@secnumdepth >\z@ \thesection \hskip1\zw\fi
749 ##1}}}%
750 \def\subsectionmark##1{\markright{%
751 \ifnum \c@secnumdepth >\@ne \thesubsection \hskip1\zw\fi
752 ##1}}}%
753 }

```

```

754 \else % if not twoside
755 \def\ps@headings{%
756 \let\@oddfoot\@empty
757 \def\@oddhead{%
758 \underline{%
759 \hbox to \fullwidth{\rightmark}\hfil\textbf{\thepage}}\hss}%
760 \let\@mkboth\markboth
761 \def\sectionmark##1{\markright{%
762 \ifnum \c@secnumdepth >\z@ \thesection \hskip1\zw\fi
763 ##1}}
764 \fi
765 %</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）。

```

766 %<*book|report>
767 \newif\if@omit@number
768 \def\ps@headings{%
769 \let\@oddfoot\@empty
770 \let\@evenfoot\@empty
771 \def\@evenhead{%
772 \if@mparswitch \hss \fi
773 \underline{\hbox to \fullwidth{\ltjsetparameter{autoxspacing={true}}
774 \textbf{\thepage}\hfil\leftmark}}}%
775 \if@mparswitch\else \hss \fi}%
776 \def\@oddhead{\underline{\hbox to \fullwidth{\ltjsetparameter{autoxspacing={true}}
777 {\if@twoside\rightmark\else\leftmark\fi}\hfil\textbf{\thepage}}}\hss}%
778 \let\@mkboth\markboth
779 \def\chaptermark##1{\markboth{%
780 \ifnum \c@secnumdepth >\m@ne
781 %<book> \if@mainmatter
782 \if@omit@number\else
783 \@chapapp\thechapter\@chappos\hskip1\zw
784 \fi
785 %<book> \fi
786 \fi
787 ##1}{}}%
788 \def\sectionmark##1{\markright{%
789 \ifnum \c@secnumdepth >\z@ \thesection \hskip1\zw\fi
790 ##1}}}%
791 %</book|report>

```

最後は学会誌の場合です。

```

792 %<*jspf>
793 \def\ps@headings{%
794 \def\@oddfoot{\normalfont\hfil\thepage\hfil}
795 \def\@evenfoot{\normalfont\hfil\thepage\hfil}
796 \def\@oddhead{\normalfont\hfil \@title \hfil}
797 \def\@evenhead{\normalfont\hfil プラズマ・核融合学会誌 \hfil}}

```

```
798 %</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。

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

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。

```
\author
\date 812 % \newcommand*{\title}[1]{\gdef\@title{#1}}
      813 % \newcommand*{\author}[1]{\gdef\@author{#1}}
      814 % \newcommand*{\date}[1]{\gdef\@date{#1}}
      815 % \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 816 %<*jspf>
\keywords 817 \newcommand*{\etitle}[1]{\gdef\@etitle{#1}}
           818 \newcommand*{\eauthor}[1]{\gdef\@eauthor{#1}}
           819 \newcommand*{\keywords}[1]{\gdef\@keywords{#1}}
           820 \newcommand*{\email}[1]{\gdef\authors@mail{#1}}
           821 \newcommand*{\AuthorsEmail}[1]{\gdef\authors@mail{author's e-mail:\ #1}}
           822 %</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。

```
823 \def\plainifnotempty{%
824   \ifx \@oddhead \@empty
```

```

825 \ifx \@oddfoot \@empty
826 \else
827 \thispagestyle{plainfoot}%
828 \fi
829 \else
830 \thispagestyle{plainhead}%
831 \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なくなり、レイアウトが変わってしまいます。

```

832 %<*article|book|report|kiyou>
833 \if@titlepage
834 \newcommand{\maketitle}{%
835 \begin{titlepage}%
836 \let\footnotesize\small
837 \let\footnoterule\relax
838 \let\footnote\thanks
839 \null\vfil
840 \if@slide
841 {\footnotesize \@date}%
842 \begin{center}
843 \mbox{} \\\[1\zw]
844 \large
845 {\maybeblue\hrule height0\jsc@mpt depth2\jsc@mpt\relax}\par
846 \jsc@smallskip
847 \@title
848 \jsc@smallskip
849 {\maybeblue\hrule height0\jsc@mpt depth2\jsc@mpt\relax}\par
850 \vfill
851 {\small \@author}%
852 \end{center}
853 \else
854 \vskip 60\jsc@mpt
855 \begin{center}%
856 {\LARGE \@title \par}%
857 \vskip 3em%
858 {\large
859 \lineskip .75em
860 \begin{tabular}[t]{c}%
861 \@author
862 \end{tabular}\par}%
863 \vskip 1.5em
864 {\large \@date \par}%
865 \end{center}%
866 \fi

```

```

867     \par
868     \@thanks\vfil\null
869 \end{titlepage}%
870 \setcounter{footnote}{0}%
871 \global\let\thanks\relax
872 \global\let\maketitle\relax
873 \global\let\@thanks\@empty
874 \global\let\@author\@empty
875 \global\let\@date\@empty
876 \global\let\@title\@empty
877 \global\let\title\relax
878 \global\let\author\relax
879 \global\let\date\relax
880 \global\let\and\relax
881 }%
882 \else
883 \newcommand{\maketitle}{\par
884 \begingroup
885 \renewcommand\thefootnote{\@fnsymbol\c@footnote}%
886 \def\@makefnmark{\rlap{\@textsuperscript{\normalfont\@thefnmark}}}%
887 \long\def\@makefntext##1{\advance\leftskip 3\zw
888 \parindent 1\zw\noindent
889 \llap{\@textsuperscript{\normalfont\@thefnmark}\hskip0.3\zw}##1}%
890 \if@twocolumn
891 \ifnum \col@number=\@ne
892 \maketitle
893 \else
894 \twocolumn[\maketitle]%
895 \fi
896 \else
897 \newpage
898 \global\@topnum\z@ % Prevents figures from going at top of page.
899 \maketitle
900 \fi
901 \plainifnotempty
902 \@thanks
903 \endgroup
904 \setcounter{footnote}{0}%
905 \global\let\thanks\relax
906 \global\let\maketitle\relax
907 \global\let\@thanks\@empty
908 \global\let\@author\@empty
909 \global\let\@date\@empty
910 \global\let\@title\@empty
911 \global\let\title\relax
912 \global\let\author\relax
913 \global\let\date\relax
914 \global\let\and\relax
915 }

```

`\@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。

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

```

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

```
998 \def\@startsection#1#2#3#4#5#6{%
999   \ifnoskipsec \leavevmode \fi
1000   \par
1001 % 見出し上の空きを \@tempskipa にセットする
1002   \@tempskipa #4\relax
1003 % \@afterindent は見出し直後の段落を字下げするかどうかを表すスイッチ
1004   \ifenglish \@afterindentfalse \else \@afterindenttrue \fi
1005 % 見出し上の空きが負なら見出し直後の段落を字下げしない
1006   \ifdim \@tempskipa <\z@
1007     \@tempskipa -\@tempskipa \@afterindentfalse
1008   \fi
1009   \if@nobreak
1010     \everypar{}%
1011   \else
1012     \addpenalty\@secpenalty
1013 % 次の行は削除
1014 %   \addvspace\@tempskipa
1015 % 次の \noindent まで追加
1016   \ifdim \@tempskipa >\z@
1017     \if@slide\else
1018       \null
1019       \vspace*{-\baselineskip}%
1020     \fi
1021     \vskip\@tempskipa
1022   \fi
1023   \fi
1024   \noindent
1025 % 追加終わり
1026   \@ifstar
1027     {\@ssect{#3}{#4}{#5}{#6}}%
1028     {\@dblarg{\@sect{#1}{#2}{#3}{#4}{#5}{#6}}}
```


`\@sect` と `\@xsect` は、前のアキがちょうどゼロの場合にもうまくいくように、多少変えてあります。

```
1029 \def\@sect#1#2#3#4#5#6[#7]#8{%
1030   \ifnum #2>\c@secnumdepth
1031     \let\@svsec\@empty
1032   \else
1033     \refstepcounter{#1}%
1034     \protected@edef\@svsec{\@secntformat{#1}\relax}%
1035   \fi
1036 % 見出し後の空きを \@tempskipa にセット
1037   \@tempskipa #5\relax
1038 % 条件判断の順序を入れ替えました
1039   \ifdim \@tempskipa<\z@
1040     \def\@svsechd{%
1041       #6{\hskip #3\relax
1042         \@svsec #8}%
1043       \csname #1mark\endcsname{#7}%
1044       \addcontentsline{toc}{#1}{%
1045         \ifnum #2>\c@secnumdepth \else
1046           \protect\numberline{\csname the#1\endcsname}%
1047         \fi
1048         #7}}% 目次にフルネームを載せるなら #8
1049   \else
1050     \begingroup
1051       \interlinepenalty \@M % 下から移動
1052       #6{%
1053         \@hangfrom{\hskip #3\relax\@svsec}%
1054       % \interlinepenalty \@M % 上に移動
1055       #8\@par}%
1056     \endgroup
1057     \csname #1mark\endcsname{#7}%
1058     \addcontentsline{toc}{#1}{%
1059       \ifnum #2>\c@secnumdepth \else
1060         \protect\numberline{\csname the#1\endcsname}%
1061       \fi
1062       #7}% 目次にフルネームを載せるならここは #8
1063   \fi
1064   \@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。

```
1065 \def\@xsect#1{%
1066 % 見出しの後ろの空きを \@tempskipa にセット
1067   \@tempskipa #1\relax
1068 % 条件判断の順序を変えました
```

```

1069 \ifdim \@tempskipa<\z@
1070   \nobreakfalse
1071   \global\@noskipsectrue
1072   \everypar{%
1073     \if@noskipsec
1074       \global\@noskipsecfalse
1075       {\setbox\z@\lastbox}%
1076       \clubpenalty\@M
1077       \begingroup \@svsechd \endgroup
1078       \unskip
1079       \@tempskipa #1\relax
1080       \hskip -\@tempskipa\ltjfakeparbegin
1081     \else
1082       \clubpenalty \@clubpenalty
1083       \everypar{}%
1084     \fi}%
1085 \else
1086   \par \nobreak
1087   \vskip \@tempskipa
1088   \@afterheading
1089 \fi
1090 \if@slide
1091   {\vskip\if@twocolumn-5\jsc@mpt\else-6\jsc@mpt\fi
1092   \maybeblue\hrule height0\jsc@mpt depth1\jsc@mpt
1093   \vskip\if@twocolumn 4\jsc@mpt\else 7\jsc@mpt\fi\relax}%
1094 \fi
1095 \par % 2000-12-18
1096 \ignorespaces}
1097 \def\@ssect#1#2#3#4#5{%
1098   \@tempskipa #3\relax
1099   \ifdim \@tempskipa<\z@
1100     \def\@svsechd{#4{\hskip #1\relax #5}}%
1101   \else
1102     \begingroup
1103     #4{%
1104       \@hangfrom{\hskip #1}%
1105       \interlinepenalty \@M #5\@@par}%
1106     \endgroup
1107   \fi
1108   \@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 本体で定義済みです。

```

\subsectionmark 1109 \newcommand*\chaptermark[1]{}
\subsubsectionmark 1110 % \newcommand*\sectionmark[1]{}
\paragraphmark 1111 % \newcommand*\subsectionmark[1]{}
\subparagraphmark 1112 % \newcommand*\subsubsectionmark[1]{}

```

```
1113 % \newcommand*{\paragraphmark}[1]{
1114 % \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。

```
1115 %<!book&!report>\setcounter{secnumdepth}{3}
1116 %<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` 1117 `\newcounter{part}`

```
1118 %<book|report>\newcounter{chapter}
```

`\c@subsubsection` 1119 %<book|report>\newcounter{section}[chapter]

`\c@paragraph` 1120 %<!book&!report>\newcounter{section}

`\c@subparagraph` 1121 \newcounter{subsection}[section]

```
1122 \newcounter{subsubsection}[subsection]
```

```
1123 \newcounter{paragraph}[subsubsection]
```

```
1124 \newcounter{subparagraph}[paragraph]
```

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

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

`\thesection` `\arabic{COUNTER}` 1, 2, 3, ...

`\thesubsection` `\roman{COUNTER}` i, ii, iii, ...

`\thesubsubsection` `\Roman{COUNTER}` I, II, III, ...

`\theparagraph` `\alph{COUNTER}` a, b, c, ...

`\thesubparagraph` `\Alph{COUNTER}` A, B, C, ...

`\kansuji{COUNTER}` 一, 二, 三, ...

以下ではスペース節約のため @ の付いた内部表現を多用しています。

```
1125 \renewcommand{\thepart}{\@Roman\c@part}
```

```
1126 %<!book&!report>% \renewcommand{\thesection}{\@arabic\c@section}
```

```
1127 %<!book&!report>\renewcommand{\thesection}{\presectionname\@arabic\c@section\postsectionname}
```

```
1128 %<!book&!report>\renewcommand{\thesubsection}{\@arabic\c@section.\@arabic\c@subsection}
```

```
1129 %<*book|report>
```

```
1130 \renewcommand{\thechapter}{\@arabic\c@chapter}
```

```
1131 \renewcommand{\thesection}{\thechapter.\@arabic\c@section}
```

```
1132 \renewcommand{\thesubsection}{\thesection.\@arabic\c@subsection}
```

```
1133 %</book|report>
```

```
1134 \renewcommand{\thesubsubsection}{%
```

```
1135 \thesubsection.\@arabic\c@subsubsection}
```

```
1136 \renewcommand{\theparagraph}{%
```

```
1137 \thesubsubsection.\@arabic\c@paragraph}
```

```
1138 \renewcommand{\thesubparagraph}{%
```

```
1139 \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。

```
1140 %<book|report>\newcommand{\@chapapp}{\prechaptername}
```

```
1141 %<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` かどうかに関らず奇数ページまで繰るように修正することで、問題を解消しました。実は、`LATEX` の標準クラスでは1998年に修正されていた問題です (コミュニティ版 `pLATEX` の標準クラス 2017/03/05 も参照)。

```
1142 %<*book>
```

```
1143 \newcommand\frontmatter{%
```

```
1144   \pltx@cleartooddpage
```

```
1145   \@mainmatterfalse
```

```
1146   \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。

```
1147 \newcommand\mainmatter{%
```

```
1148   \pltx@cleartooddpage
```

```
1149   \@mainmattertrue
```

```
1150   \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。

```
1151 \newcommand\backmatter{%
```

```
1152   \if@openleft
```

```
1153     \cleardoublepage
```

```
1154   \else\if@openright
```

```
1155     \cleardoublepage
```

```
1156   \else
```

```
1157     \clearpage
```

```
1158   \fi\fi
```

```
1159   \@mainmatterfalse}
```

```
1160 %</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。

```
1161 %<!*book&!report>
1162 \newcommand\part{%
1163   \if@noskipsec \leavevmode \fi
1164   \par
1165   \addvspace{4ex}%
1166   \if@english \@afterindentfalse \else \@afterindenttrue \fi
1167   \secdef\@part\@spart}
1168 %</!*book&!report>
```

`book` および `report` クラスの場合は、少し複雑です。

```
1169 %<*book|report>
1170 \newcommand\part{%
1171   \if@openleft
1172     \cleardoublepage
1173   \else\if@openright
1174     \cleardoublepage
1175   \else
1176     \clearpage
1177   \fi\fi
1178   \thispagestyle{empty}% 欧文用標準スタイルでは plain
1179   \if@twocolumn
1180     \onecolumn
1181     \@restonecoltrue
1182   \else
1183     \@restonecolfalse
1184   \fi
1185   \null\vfil
1186   \secdef\@part\@spart}
1187 %</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。

```
1188 %<!*book&!report>
1189 \def\@part [#1]#2{%
1190   \ifnum \c@secnumdepth >\m@ne
1191     \refstepcounter{part}%
1192     \addcontentsline{toc}{part}{%
1193       \prepartname\thepart\postpartname\hspace{1\zw}#1}%
1194   \else
```

```

1195 \addcontentsline{toc}{part}{#1}%
1196 \fi
1197 \markboth{}{}%
1198 {\parindent\z@
1199 \raggedright
1200 \interlinepenalty \@M
1201 \normalfont
1202 \ifnum \c@secnumdepth >\m@ne
1203 \Large\headfont\prepartname\thepart\postpartname
1204 \par\nobreak
1205 \fi
1206 \huge \headfont #2%
1207 \markboth{}{}\par}%
1208 \nobreak
1209 \vskip 3ex
1210 \@afterheading}
1211 %</!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。

```

1212 %<*book|report>
1213 \def\@part[#1]#2{%
1214 \ifnum \c@secnumdepth >-2\relax
1215 \refstepcounter{part}%
1216 \addcontentsline{toc}{part}{%
1217 \prepartname\thepart\postpartname\hspace{1\zw}#1}%
1218 \else
1219 \addcontentsline{toc}{part}{#1}%
1220 \fi
1221 \markboth{}{}%
1222 {\centering
1223 \interlinepenalty \@M
1224 \normalfont
1225 \ifnum \c@secnumdepth >-2\relax
1226 \huge\headfont \prepartname\thepart\postpartname
1227 \par\vskip20\jsc@mp
1228 \fi
1229 \Huge \headfont #2\par}%
1230 \@endpart}
1231 %</book|report>

```

\@spart 番号を付けない部です。

```

1232 %<*!book&!report>
1233 \def\@spart#1{%
1234 \parindent \z@ \raggedright
1235 \interlinepenalty \@M
1236 \normalfont
1237 \huge \headfont #1\par}%
1238 \nobreak
1239 \vskip 3ex

```

```

1240 \afterheading}
1241 %<!book&!report>
1242 %<*book|report>
1243 \def\@spart#1{%
1244     \centering
1245     \interlinepenalty \@M
1246     \normalfont
1247     \Huge \headfont #1\par}%
1248 \@endpart}
1249 %</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。

```

1250 %<*book|report>
1251 \def\@endpart{\vfil\newpage
1252     \if@twoside
1253     \if@openleft %% added (2017/02/24)
1254     \null\thispagestyle{empty}\newpage
1255     \else\if@openright %% added (2016/12/13)
1256     \null\thispagestyle{empty}\newpage
1257     \fi\fi %% added (2016/12/13, 2017/02/24)
1258     \fi
1259     \if@restonecol
1260     \twocolumn
1261     \fi}
1262 %</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。

```

1263 %<*book|report>
1264 \newcommand{\chapter}{%
1265     \if@openleft\cleardoublepage\else
1266     \if@openright\cleardoublepage\else\clearpage\fi\fi
1267     \plainifnotempty % 元: \thispagestyle{plain}
1268     \global\@topnum\z@
1269     \if@english \@afterindentfalse \else \@afterindenttrue \fi
1270     \secdef
1271     {\@omit@numberfalse\@chapter}%
1272     {\@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。

```

1273 \def\@chapter[#1]#2{%
1274   \ifnum \c@secnumdepth >\m@ne
1275 %<book>   \if@mainmatter
1276     \refstepcounter{chapter}%
1277     \typeout{\@chapapp\thechapter\@chappos}%
1278     \addcontentsline{toc}{chapter}%
1279       {\protect\numberline
1280        % {\if@english\thechapter\else\@chapapp\thechapter\@chappos\fi}%
1281        {\@chapapp\thechapter\@chappos}%
1282        #1}%
1283 %<book>   \else\addcontentsline{toc}{chapter}{#1}\fi
1284 \else
1285   \addcontentsline{toc}{chapter}{#1}%
1286 \fi
1287 \chaptermark{#1}%
1288 \addtocontents{lof}{\protect\addvspace{10\jsc@empt}}%
1289 \addtocontents{lot}{\protect\addvspace{10\jsc@empt}}%
1290 \if@twocolumn
1291   \@topnewpage[\@makechapterhead{#2}]%
1292 \else
1293   \@makechapterhead{#2}%
1294   \@afterheading
1295 \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。

```

1296 \def\@makechapterhead#1{%
1297   \vspace*{2\Cvs}% 欧文は 50pt
1298   {\parindent \z@ \raggedright \normalfont
1299     \ifnum \c@secnumdepth >\m@ne
1300 %<book>     \if@mainmatter
1301       \huge\headfont \@chapapp\thechapter\@chappos
1302       \par\nobreak
1303       \vskip \Cvs % 欧文は 20pt
1304 %<book>     \fi
1305   \fi
1306   \interlinepenalty\@M
1307   \Huge \headfont #1\par\nobreak
1308   \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。

```

1309 \def\@schapter#1{%
1310   \chaptermark{#1}%
1311   \if@twocolumn
1312     \@topnewpage[\@makeschapterhead{#1}]%
1313   \else
1314     \@makeschapterhead{#1}\@afterheading
1315   \fi}

```

`\@makeschapterhead` 番号なしの章見出しです。


```

1316 \def\@makeschapterhead#1{%
1317   \vspace*{2\Cvs}% 欧文は 50pt
1318   {\parindent \z@ \raggedright
1319     \normalfont
1320     \interlinepenalty\@M
1321     \Huge \headfont #1\par\nobreak
1322     \vskip 3\Cvs}} % 欧文は 40pt
1323 %</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。

```

1324 \if@twocolumn
1325   \newcommand{\section}{%
1326 %<jspf>\ifx\maketitle\relax\else\maketitle\fi
1327   \@startsection{section}{1}{\z@}%
1328 %<!kiyou>   {0.6\Cvs}{0.4\Cvs}%
1329 %<kiyou>    {\Cvs}{0.5\Cvs}%
1330 %   {\normalfont\large\headfont\@secapp}}
1331   {\normalfont\large\headfont\raggedright}}
1332 \else
1333   \newcommand{\section}{%
1334     \if@slide\clearpage\fi
1335     \@startsection{section}{1}{\z@}%
1336     {\Cvs \@plus.5\Cdp \@minus.2\Cdp}% 前アキ
1337     {.5\Cvs \@plus.3\Cdp}% 後アキ
1338 %   {\normalfont\Large\headfont\@secapp}}
1339     {\normalfont\Large\headfont\raggedright}}
1340 \fi

```

`\subsection` 同上です。

```

1341 \if@twocolumn
1342   \newcommand{\subsection}{\@startsection{subsection}{2}{\z@}%
1343     {\z@}{\if@slide .4\Cvs \else \z@ \fi}%
1344     {\normalfont\normalsize\headfont}}
1345 \else
1346   \newcommand{\subsection}{\@startsection{subsection}{2}{\z@}%
1347     {\Cvs \@plus.5\Cdp \@minus.2\Cdp}% 前アキ
1348     {.5\Cvs \@plus.3\Cdp}% 後アキ
1349     {\normalfont\large\headfont}}
1350 \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)。

```

1351 \if@twocolumn
1352   \newcommand{\subsubsection}{\@startsection{subsubsection}{3}{\z@}%

```

```

1353     {\z@}{\if@slide .4\Cvs \else \z@ \fi}%
1354     {\normalfont\normalsize\headfont}}
1355 \else
1356   \newcommand{\subsubsection}{\@startsection{subsubsection}{3}{\z@}%
1357     {\Cvs \@plus.5\Cdp \@minus.2\Cdp}%
1358     {\if@slide .5\Cvs \@plus.3\Cdp \else \z@ \fi}%
1359     {\normalfont\normalsize\headfont}}
1360 \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。

```

1361 %<!jspf>\newcommand{\jsParagraphMark}{■}
1362 \if@twocolumn
1363   \newcommand{\paragraph}{\@startsection{paragraph}{4}{\z@}%
1364     {\z@}{\if@slide .4\Cvs \else -1\zw\fi}% 改行せず 1\zw のアキ
1365 %<jspf>   {\normalfont\normalsize\headfont}}
1366 %<!jspf>   {\normalfont\normalsize\headfont\jsParagraphMark}}
1367 \else
1368   \newcommand{\paragraph}{\@startsection{paragraph}{4}{\z@}%
1369     {0.5\Cvs \@plus.5\Cdp \@minus.2\Cdp}%
1370     {\if@slide .5\Cvs \@plus.3\Cdp \else -1\zw\fi}% 改行せず 1\zw のアキ
1371 %<jspf>   {\normalfont\normalsize\headfont}}
1372 %<!jspf>   {\normalfont\normalsize\headfont\jsParagraphMark}}
1373 \fi

```

`\subparagraph` 見出しの後ろで改行されません。

```

1374 \if@twocolumn
1375   \newcommand{\subparagraph}{\@startsection{subparagraph}{5}{\z@}%
1376     {\z@}{\if@slide .4\Cvs \@plus.3\Cdp \else -1\zw\fi}%
1377     {\normalfont\normalsize\headfont}}
1378 \else
1379   \newcommand{\subparagraph}{\@startsection{subparagraph}{5}{\z@}%
1380     {\z@}{\if@slide .5\Cvs \@plus.3\Cdp \else -1\zw\fi}%
1381     {\normalfont\normalsize\headfont}}
1382 \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 `\leftmargink` に設定します。

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

[2002-05-11] 3\zw に変更しました。

[2005-03-19] 二段組は 2\zw に戻しました。

```
1383 \if@slide
1384 \setlength\leftmargini{1\zw}
1385 \else
1386 \if@twocolumn
1387 \setlength\leftmargini{2\zw}
1388 \else
1389 \setlength\leftmargini{3\zw}
1390 \fi
1391 \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 1392 \if@slide
\leftmarginv 1393 \setlength\leftmarginii {1\zw}
\leftmarginvi 1394 \setlength\leftmarginiii{1\zw}
1395 \setlength\leftmarginiv {1\zw}
1396 \setlength\leftmarginv {1\zw}
1397 \setlength\leftmarginvi {1\zw}
1398 \else
1399 \setlength\leftmarginii {2\zw}
1400 \setlength\leftmarginiii{2\zw}
1401 \setlength\leftmarginiv {2\zw}
1402 \setlength\leftmarginv {1\zw}
1403 \setlength\leftmarginvi {1\zw}
1404 \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。

```
1405 \setlength \labelsep {0.5\zw} % .5em
1406 \setlength \labelwidth{\leftmargini}
1407 \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。

```
1408 \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 1409 \@beginparpenalty -\@lowpenalty
\@itempenalty 1410 \@endparpenalty -\@lowpenalty
1411 \@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 $_{-0.1}^{+0.2}$ `\baselineskip` を思い切って外しました。

```
1412 \def\@listi{\leftmargin\leftmarginI
1413 \parsep \z@
1414 \topsep 0.5\baselineskip
1415 \itemsep \z@ \relax}
1416 \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)。

```
1417 \@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 1418 \def\@listii{\leftmargin\leftmarginii
\@listiv 1419 \labelwidth\leftmarginii \advance\labelwidth-\labelsep
1420 \topsep \z@
\@listv 1421 \parsep \z@
\@listvi 1422 \itemsep\parsep}
1423 \def\@listiii{\leftmargin\leftmarginiii
1424 \labelwidth\leftmarginiii \advance\labelwidth-\labelsep
1425 \topsep \z@
1426 \parsep \z@
1427 \itemsep\parsep}
1428 \def\@listiv {\leftmargin\leftmarginiv
1429 \labelwidth\leftmarginiv
1430 \advance\labelwidth-\labelsep}
1431 \def\@listv {\leftmargin\leftmarginv
1432 \labelwidth\leftmarginv
1433 \advance\labelwidth-\labelsep}
1434 \def\@listvi {\leftmargin\leftmarginvi
1435 \labelwidth\leftmarginvi
1436 \advance\labelwidth-\labelsep}
```

■**enumerate 環境** `enumerate` 環境はカウンタ `enumi`, `enumii`, `enumiii`, `enumiv` を使います。`enumn` は第 n レベルの番号です。

`\theenumi` 出力する番号の書式を設定します。これらは L^AT_EX 本体 (`ltlists.dtx` 参照) で定義済みですが、ここでは表し方を変えています。`\@arabic`, `\@alph`, `\@roman`, `\@Alph` はそれぞれ算用数字, 小文字アルファベット, 小文字ローマ数字, 大文字アルファベットで番号を出力する命令です。

```
1437 \renewcommand{\theenumi}{\@arabic\c@enumi}
1438 \renewcommand{\theenumii}{\@alph\c@enumii}
1439 \renewcommand{\theenumiii}{\@roman\c@enumiii}
1440 \renewcommand{\theenumiv}{\@Alph\c@enumiv}
```

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

```

\labelenumiv 1441 \newcommand{\labelenumi}{\theenumi.}
1442 \newcommand{\labelenumii}{\inhibitglue (\theenumii) \inhibitglue}
1443 \newcommand{\labelenumiii}{\theenumiii.}
1444 \newcommand{\labelenumiv}{\theenumiv.}

```

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

```

\p@enumiv 1445 \renewcommand{\p@enumii}{\theenumi}
1446 \renewcommand{\p@enumiii}{\theenumi\inhibitglue (\theenumii) }
1447 \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 1448 \newcommand\labelitemi{\textbullet}
\labelitemiii 1449 \newcommand\labelitemii{\normalfont\bfseries \textendash}
1450 \newcommand\labelitemiii{\textasteriskcentered}
\labelitemiv 1451 \newcommand\labelitemiv{\textperiodcentered}

```

■description 環境

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

```

1452 \newenvironment{description}{%
1453 \list{}{%
1454 \labelwidth=\leftmargin
1455 \labelsep=1\zw
1456 \advance \labelwidth by -\labelsep
1457 \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。

```

1458 \newcommand*\descriptionlabel[1]{\normalfont\headfont #1\hfil}

```

■概要

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

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

```

1459 %<*book>
1460 \newenvironment{abstract}{%

```

```

1461 \begin{list}{}{%
1462   \listparindent=1\zw
1463   \itemindent=\listparindent
1464   \rightmargin=0pt
1465   \leftmargin=5\zw}\item[]{}\end{list}\vspace{\baselineskip}}
1466 %</book>
1467 %<*article|report|kiyou>
1468 \newbox\@abstractbox
1469 \if@titlepage
1470   \newenvironment{abstract}{%
1471     \titlepage
1472     \null\vfil
1473     \@beginparpenalty\@lowpenalty
1474     \begin{center}%
1475       \headfont \abstractname
1476       \@endparpenalty\@M
1477     \end{center}}%
1478   {\par\vfil\null\endtitlepage}
1479 \else
1480   \newenvironment{abstract}{%
1481     \if@twocolumn
1482       \ifx\maketitle\relax
1483         \section*{\abstractname}%
1484       \else
1485         \global\setbox\@abstractbox\hbox\bgroup
1486         \begin{minipage}[b]{\textwidth}
1487           \small\parindent1\zw
1488           \begin{center}%
1489             {\headfont \abstractname\vspace{-.5em}\vspace{\z@}}%
1490           \end{center}%
1491           \list{}{%
1492             \listparindent\parindent
1493             \itemindent \listparindent
1494             \rightmargin \leftmargin}%
1495           \item\relax
1496         \fi
1497       \else
1498         \small
1499         \begin{center}%
1500           {\headfont \abstractname\vspace{-.5em}\vspace{\z@}}%
1501         \end{center}%
1502         \list{}{%
1503           \listparindent\parindent
1504           \itemindent \listparindent
1505           \rightmargin \leftmargin}%
1506         \item\relax
1507       \fi}\if@twocolumn
1508     \ifx\maketitle\relax
1509     \else

```

```

1510         \endlist\end{minipage}\egroup
1511     \fi
1512 \else
1513     \endlist
1514 \fi}
1515 \fi
1516 %</article|report|kiyou>
1517 %<*jspf>
1518 \newbox\@abstractbox
1519 \newenvironment{abstract}{%
1520 \global\setbox\@abstractbox\hbox\bgroup
1521 \begin{minipage}[b]{157\jsc@mmm}{\sffamily Abstract}\par
1522 \small
1523 \if@english \parindent6\jsc@mmm \else \parindent1\zw \fi}%
1524 {\end{minipage}\egroup}
1525 %</jspf>

```

■キーワード

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

```

1526 %<*jspf>
1527 %\newbox\@keywordsbox
1528 %\newenvironment{keywords}{%
1529 % \global\setbox\@keywordsbox\hbox\bgroup
1530 % \begin{minipage}[b]{157\jsc@mmm}{\sffamily Keywords:}\par
1531 % \small\parindent0\zw}%
1532 % {\end{minipage}\egroup}
1533 %</jspf>

```

■verse 環境

verse 詩のための `verse` 環境です。

```

1534 \newenvironment{verse}{%
1535 \let \=\@centercr
1536 \list{}{%
1537 \itemsep \z@
1538 \itemindent -2\zw % 元: -1.5em
1539 \listparindent\itemindent
1540 \rightmargin \z@
1541 \advance\leftmargin 2\zw}% 元: 1.5em
1542 \item\relax}\endlist}

```

■quotation 環境

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

```

1543 \newenvironment{quotation}{%
1544 \list{}{%

```

```

1545 \listparindent\parindent
1546 \itemindent\listparindent
1547 \rightmargin \z@}%
1548 \item\relax}{\endlist}

```

■quote 環境

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

```

1549 \newenvironment{quote}%
1550 {\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。

```

1551 \def\@begintheorem#1#2{\trivlist\labelsep=1\zw
1552 \item[\hskip \labelsep{\headfont #1\ #2}]}
1553 \def\@opargbegintheorem#1#2#3{\trivlist\labelsep=1\zw
1554 \item[\hskip \labelsep{\headfont #1\ #2 (#3)}]}

```

titlepage タイトルを独立のページに出力するのに使われます。

[2017-02-24] コミュニティ版 p_AT_EX の標準クラス 2017/02/15 に合わせて、book クラスでタイトルを必ず奇数ページに送るようにしました。といっても、横組クラスしかありませんでしたので、従来の挙動は何も変わっていません。また、book 以外の場合のページ番号のリセットもコミュニティ版 p_AT_EX の標準クラス 2017/02/15 に合わせましたが、こちら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。

```

1555 \newenvironment{titlepage}{%
1556 %<book> \pltx@cleartooddpage %% 2017-02-24
1557 \if@twocolumn
1558 \@restonecoltrue\onecolumn
1559 \else
1560 \@restonecolfalse\newpage
1561 \fi
1562 \thispagestyle{empty}%
1563 \ifodd\c@page\setcounter{page}\@ne\else\setcounter{page}\z@\fi
1564 }%
1565 {\if@restonecol\twocolumn \else \newpage \fi
1566 \if@twoside\else
1567 \setcounter{page}\@ne
1568 \fi}

```


■付録

`\appendix` 本文と付録を分離するコマンドです。

```
1569 %<!*book&!report>
1570 \newcommand{\appendix}{\par
1571 \setcounter{section}{0}%
1572 \setcounter{subsection}{0}%
1573 \gdef\presectionname{\appendixname}%
1574 \gdef\postsectionname{}}%
1575 % \gdef\thesection{\@Alph\c@section}% [2003-03-02]
1576 \gdef\thesection{\presectionname\@Alph\c@section\postsectionname}%
1577 \gdef\thesubsection{\@Alph\c@section.\@arabic\c@subsection}}
1578 %</!*book&!report>
1579 %<*book|report>
1580 \newcommand{\appendix}{\par
1581 \setcounter{chapter}{0}%
1582 \setcounter{section}{0}%
1583 \gdef\@chapapp{\appendixname}%
1584 \gdef\@chappos{}}%
1585 \gdef\thechapter{\@Alph\c@chapter}}
1586 %</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。

```
1587 \setlength\arraycolsep{5\jsc@mpt}
```

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

```
1588 \setlength\tabcolsep{6\jsc@mpt}
```

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

```
1589 \setlength\arrayrulewidth{.4\jsc@mpt}
```

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

```
1590 \setlength\doublerulesep{2\jsc@mpt}
```

■tabbing 環境

`\tabbingsep` \' コマンドで入るアキです。

```
1591 \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。

```
1592 \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。

```
1593 \setlength\fboxsep{3\jsc@mpt}
```

```
1594 \setlength\fboxrule{.4\jsc@mpt}
```

■equation と eqnarray 環境

`\theequation` 数式番号を出力するコマンドです。

```
1595 %<!book&!report>\renewcommand \theequation {\@arabic\c@equation}
```

```
1596 %<*book|report>
```

```
1597 \@addtoreset{equation}{chapter}
```

```
1598 \renewcommand\theequation
```

```
1599 {\ifnum \c@chapter>\z@ \thechapter.\fi \@arabic\c@equation}
```

```
1600 %</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。

```
1601 % \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。

```
1602 % \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。

```
1603 % \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。

```
1604 %<!*book&!report>
1605 \newcounter{figure}
1606 \renewcommand \thefigure {\@arabic\c@figure}
1607 %</!*book&!report>
1608 %<*book|report>
1609 \newcounter{figure}[chapter]
1610 \renewcommand \thefigure
1611     {\ifnum \c@chapter>\z@ \thechapter.\fi \@arabic\c@figure}
1612 %</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 1613 \def\fps@figure{tbp}
1614 \def\ftype@figure{1}
\fnun@figure 1615 \def\ext@figure{lof}
1616 \def\fnun@figure{\figurename\nobreak\thefigure}
```

`figure` * 形式は段抜きフロートです。

```
figure* 1617 \newenvironment{figure}%
1618     {\@float{figure}}%
1619     {\end@float}
1620 \newenvironment{figure*}%
1621     {\@dblfloat{figure}}%
1622     {\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。

```
1623 %<!*book&!report>
1624 \newcounter{table}
1625 \renewcommand\thetable{\@arabic\c@table}
1626 %</!*book&!report>
1627 %<*book|report>
1628 \newcounter{table}[chapter]
1629 \renewcommand \thetable
1630     {\ifnum \c@chapter>\z@ \thechapter.\fi \@arabic\c@table}
1631 %</book|report>
```

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

```
\ext@table 1632 \def\fps@table{tbp}
1633 \def\ftype@table{2}
\fnun@table 1634 \def\ext@table{lot}
1635 \def\fnun@table{\tablename\nobreak\thetable}
```

`table` * は段抜きフロートです。

`table*`

```

1636 \newenvironment{table}%
1637     {\@float{table}}%
1638     {\end@float}
1639 \newenvironment{table*}%
1640     {\@dblfloat{table}}%
1641     {\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。

```

1642 \newlength\abovecaptionskip
1643 \newlength\belowcaptionskip
1644 \setlength\abovecaptionskip{5\jsc@empt} % 元: 10\p@
1645 \setlength\belowcaptionskip{5\jsc@empt} % 元: 0\p@

```

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

[2003-11-05] ロジックを少し変えてみました。

[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.
```

```

1646 %<*\jspf>
1647 % \long\def\@makecaption#1#2{\small
1648 %   \advance\leftskip10\jsc@mmm
1649 %   \advance\rightskip10\jsc@mmm
1650 %   \vskip\abovecaptionskip
1651 %   \sbox\@tempboxa{#1{\hskip1\zw}#2}%
1652 %   \ifdim \wd\@tempboxa >\hsize
1653 %     #1{\hskip1\zw}#2\par
1654 %   \else
1655 %     \global \@minipagefalse
1656 %     \hb@xt@\hsize{\hfil\box\@tempboxa\hfil}%
1657 %   \fi
1658 %   \vskip\belowcaptionskip}}
1659 \long\def\@makecaption#1#2{\small
1660   \advance\leftskip .0628\linewidth
1661   \advance\rightskip .0628\linewidth
1662   \vskip\abovecaptionskip
1663   \sbox\@tempboxa{#1{\hskip1\zw}#2}%
1664   \ifdim \wd\@tempboxa <\hsize \centering \fi

```

```

1665 #1{\hskip1\zw}#2\par
1666 \vskip\belowcaptionskip}}
1667 %</!jspf>
1668 %<*jspf>
1669 \long\def\makecaption#1#2{%
1670 \vskip\abovecaptionskip
1671 \sbox\@tempboxa{\small\sffamily #1\quad #2}%
1672 \ifdim \wd\@tempboxa >\hsize
1673   {\small\sffamily
1674     \list{#1}{%
1675       \renewcommand{\makelabel}[1]{##1\hfil}
1676       \itemsep \z@
1677       \itemindent \z@
1678       \labelsep \z@
1679       \labelwidth 11\jsc@mmm
1680       \listparindent\z@
1681       \leftmargin 11\jsc@mmm}\item\relax #2\endlist}
1682 \else
1683   \global \@minipagefalse
1684   \hb@xt@\hsize{\hfil\box\@tempboxa\hfil}%
1685 \fi
1686 \vskip\belowcaptionskip}
1687 %</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
\jsc@warnoldfontcmdexception 1688 \newif\if@jsc@warnoldfontcmd
1689 \@jsc@warnoldfontcmdtrue
1690 \newif\if@jsc@warnoldfontcmdexception
1691 \@jsc@warnoldfontcmdexceptionfalse

\jsc@DeclareOldFontCommand
1692 \newcommand*{\jsc@DeclareOldFontCommand}[3]{%
1693   \DeclareOldFontCommand{#1}{%
1694     \jsc@warnoldfontcmd{#1}#2%
1695   }{%
1696     \jsc@warnoldfontcmd{#1}#3%

```

```

1697 }%
1698 }
1699 \DeclareRobustCommand*\jsc@warnoldfontcmd}[1]{%
1700 \if@jsc@warnoldfontcmdexception\else\if@jsc@warnoldfontcmd
1701 \ClassWarning{\jsc@clsname}{%
1702   deprecated old font command '\string#1' used.\MessageBreak
1703   You should note, that since 1994 LaTeX2e provides a\MessageBreak
1704   new font selection scheme called NFSS2 with several\MessageBreak
1705   new, combinable font commands. This \jsc@clsname\MessageBreak
1706   class has defined the old font commands like\MessageBreak
1707   '\string#1' only for compatibility%
1708 }%
1709 \global\@jsc@warnoldfontcmdfalse
1710 \fi\fi
1711 }

```

`\mc` フォントファミリーを変更します。

```

\gt 1712 \jsc@DeclareOldFontCommand{\mc}{\normalfont\mcfamily}{\mathmc}
\rm 1713 \jsc@DeclareOldFontCommand{\gt}{\normalfont\gtfamily}{\mathgt}
\s1 1714 \jsc@DeclareOldFontCommand{\rm}{\normalfont\rmfamily}{\mathrm}
\s1 1715 \jsc@DeclareOldFontCommand{\sf}{\normalfont\sffamily}{\mathsf}
\tt 1716 \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。

```
1717 \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せん (警告メッセージを出力します)。通常のアップライト体に戻すコマンドは `\upshape` です。

```

\it 1718 \jsc@DeclareOldFontCommand{\it}{\normalfont\itshape}{\mathit}
\s1 1719 \jsc@DeclareOldFontCommand{\sl}{\normalfont\slshape}{\@nomath\sl}
\sc 1720 \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 1721 \DeclareRobustCommand*\cal}{\@fontswitch\relax\mathcal}
1722 \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。

```
1723 \newcommand\@pnumwidth{1.55em}
1724 \newcommand\@tocrmarg{2.55em}
1725 \newcommand\@dotsep{4.5}
1726 %<!book&!report>\setcounter{tocdepth}{2}
1727 %<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)

```
1728 \newdimen\jsc@tocl@width
1729 \newcommand{\tableofcontents}{%
1730 %<*book|report>
1731 \settowidth\jsc@tocl@width{\headfont\prechaptername\postchaptername}%
1732 \settowidth\@tempdima{\headfont\appendixname}%
1733 \ifdim\jsc@tocl@width<\@tempdima \setlength\jsc@tocl@width{\@tempdima}\fi
1734 \ifdim\jsc@tocl@width<2\zw \divide\jsc@tocl@width by 2 \advance\jsc@tocl@width 1\zw\fi
```

```

1735 \if@twocolumn
1736   \@restonecoltrue\onecolumn
1737 \else
1738   \@restonecolfalse
1739 \fi
1740 \chapter*{\contentsname}%
1741 \@mkboth{\contentsname}{}%
1742 %</book|report>
1743 %<!*book&!report>
1744 \settowidth\jsc@tocl@width{\headfont\presectionname\postsectionname}%
1745 \settowidth\@tempdima{\headfont\appendixname}%
1746 \ifdim\jsc@tocl@width<\@tempdima\relax\setlength\jsc@tocl@width{\@tempdima}\fi
1747 \ifdim\jsc@tocl@width<2\zw \divide\jsc@tocl@width by 2 \advance\jsc@tocl@width 1\zw\fi
1748 \section*{\contentsname}%
1749 \@mkboth{\contentsname}{\contentsname}%
1750 %</!book&!report>
1751 \@starttoc{toc}%
1752 %<book|report> \if@restonecol\twocolumn\fi
1753 }

```

\l@part 部の目次です。

```

1754 \newcommand*{\l@part}[2]{%
1755   \ifnum \c@tocdepth >-2\relax
1756   %<!book&!report> \addpenalty\@secpenalty
1757   %<book|report> \addpenalty{-\@highpenalty}%
1758   \addvspace{2.25em \@plus\jsc@empt}%
1759   \begingroup
1760     \parindent \z@
1761   % \@pnumwidth should be \@tocrmarg
1762   % \rightskip \@pnumwidth
1763   % \rightskip \@tocrmarg
1764   %\parfillskip -\rightskip
1765   {\leavevmode
1766     \large \headfont
1767     \setlength\@lnumwidth{4\zw}%
1768     #1\hfil \hb@xt@\@pnumwidth{\hss #2}}\par
1769   \nobreak
1770 %<book|report> \global\@nobreaktrue
1771 %<book|report> \everypar{\global\@nobreakfalse\everypar{}}%
1772   \endgroup
1773   \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)

```

1774 %<*book|report>
1775 \newcommand*{\l@chapter}[2]{%
1776   \ifnum \c@tocdepth >\m@ne
1777     \addpenalty{-\@highpenalty}%

```



```

1778 \addvspace{1.0em \@plus\jsc@mp}
1779 % \vskip 1.0em \@plus\p@ % book.cls では↑がこうなっている
1780 \begingroup
1781 \parindent\z@
1782 % \rightskip\@pnumwidth
1783 \rightskip\@tocrmarg
1784 \parfillskip-\rightskip
1785 \leavevmode\headfont
1786 % \if@english\setlength\@lnumwidth{5.5em}\else\setlength\@lnumwidth{4.683\zw}\fi
1787 \setlength\@lnumwidth{\jsc@tocl@width}\advance\@lnumwidth 2.683\zw
1788 \advance\leftskip\@lnumwidth \hskip-\leftskip
1789 #1\nobreak\hfil\nobreak\hbox to\@pnumwidth{\hss#2}\par
1790 \penalty\@highpenalty
1791 \endgroup
1792 \fi}
1793 %</book|report>

```

`\l@section` 節の目次です。

```

1794 %<!*book&!report>
1795 \newcommand*{\l@section}[2]{%
1796 \ifnum \c@tocdepth >\z@
1797 \addpenalty{\@secpenalty}%
1798 \addvspace{1.0em \@plus\jsc@mp}%
1799 \begingroup
1800 \parindent\z@
1801 % \rightskip\@pnumwidth
1802 \rightskip\@tocrmarg
1803 \parfillskip-\rightskip
1804 \leavevmode\headfont
1805 %\setlength\@lnumwidth{4\zw}% 元 1.5em [2003-03-02]
1806 \setlength\@lnumwidth{\jsc@tocl@width}\advance\@lnumwidth 2\zw
1807 \advance\leftskip\@lnumwidth \hskip-\leftskip
1808 #1\nobreak\hfil\nobreak\hbox to\@pnumwidth{\hss#2}\par
1809 \endgroup
1810 \fi}
1811 %</!*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。

```

1812 <book|report> % \newcommand*{\l@section}{\dottedtocline{1}{1\zw}{3.683\zw}}
[2013-12-30] 上のインデントは \jsc@tocl@width から決めるようにしました。(by ts)

```

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

`\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 1813 %<!*book&!report>
1814 % \newcommand*{\l@subsection} {\dottedtocline{2}{1.5em}{2.3em}}
1815 % \newcommand*{\l@subsubsection}{\dottedtocline{3}{3.8em}{3.2em}}
1816 % \newcommand*{\l@paragraph} {\dottedtocline{4}{7.0em}{4.1em}}

```

```

1817 % \newcommand*\l@subparagraph {\@dottedtocline{5}{10em}{5em}}
1818 %
1819 % \newcommand*\l@subsection {\@dottedtocline{2}{1\zw}{3\zw}}
1820 % \newcommand*\l@subsubsection {\@dottedtocline{3}{2\zw}{3\zw}}
1821 % \newcommand*\l@paragraph {\@dottedtocline{4}{3\zw}{3\zw}}
1822 % \newcommand*\l@subparagraph {\@dottedtocline{5}{4\zw}{3\zw}}
1823 %
1824 \newcommand*\l@subsection{%
1825     \@tempdima\jsc@tocl@width \advance\@tempdima -1\zw
1826     \@dottedtocline{2}{\@tempdima}{3\zw}}
1827 \newcommand*\l@subsubsection{%
1828     \@tempdima\jsc@tocl@width \advance\@tempdima 0\zw
1829     \@dottedtocline{3}{\@tempdima}{4\zw}}
1830 \newcommand*\l@paragraph{%
1831     \@tempdima\jsc@tocl@width \advance\@tempdima 1\zw
1832     \@dottedtocline{4}{\@tempdima}{5\zw}}
1833 \newcommand*\l@subparagraph{%
1834     \@tempdima\jsc@tocl@width \advance\@tempdima 2\zw
1835     \@dottedtocline{5}{\@tempdima}{6\zw}}
1836 %</!book&!report>
1837 %<*book|report>
1838 % \newcommand*\l@subsection {\@dottedtocline{2}{3.8em}{3.2em}}
1839 % \newcommand*\l@subsubsection {\@dottedtocline{3}{7.0em}{4.1em}}
1840 % \newcommand*\l@paragraph {\@dottedtocline{4}{10em}{5em}}
1841 % \newcommand*\l@subparagraph {\@dottedtocline{5}{12em}{6em}}
1842 \newcommand*\l@section{%
1843     \@tempdima\jsc@tocl@width \advance\@tempdima -1\zw
1844     \@dottedtocline{1}{\@tempdima}{3.683\zw}}
1845 \newcommand*\l@subsection{%
1846     \@tempdima\jsc@tocl@width \advance\@tempdima 2.683\zw
1847     \@dottedtocline{2}{\@tempdima}{3.5\zw}}
1848 \newcommand*\l@subsubsection{%
1849     \@tempdima\jsc@tocl@width \advance\@tempdima 6.183\zw
1850     \@dottedtocline{3}{\@tempdima}{4.5\zw}}
1851 \newcommand*\l@paragraph{%
1852     \@tempdima\jsc@tocl@width \advance\@tempdima 10.683\zw
1853     \@dottedtocline{4}{\@tempdima}{5.5\zw}}
1854 \newcommand*\l@subparagraph{%
1855     \@tempdima\jsc@tocl@width \advance\@tempdima 16.183\zw
1856     \@dottedtocline{5}{\@tempdima}{6.5\zw}}
1857 %</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 ですが、アスキー版では \@tempdima の代わりに \@lnumwidth という変数で幅を決めるよう
 に再定義しています。後続文字が全角か半角かでスペースが変わらないように \hspace を
 入れておきました。

```

1858 \newdimen\@lnumwidth
1859 \def\numberline#1{\hb@xt@\@lnumwidth{#1\hfil}\hspace{0pt}}

```

`\dottedtocline` L^AT_EX 本体 (l^AT_EX 2.09 参照) での定義と同じですが、`\@tempdima` を `\@lnumwidth` に
`\jsTocLine` 変えています。

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

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

とします。

```
1860 \def\jsTocLine{\leaders\hbox{%
1861   $\m@th \mkern \@dotsep mu\hbox{.}\mkern \@dotsep mu$\}\hfill}
1862 \def\dottedtocline#1#2#3#4#5{\ifnum #1>\c@tocdepth \else
1863   \vskip \z@ \@plus.2\jsc@mpt
1864   {\leftskip #2\relax \rightskip \@tocrmarg \parfillskip -\rightskip
1865     \parindent #2\relax\@afterindenttrue
1866     \interlinepenalty\@M
1867     \leavevmode
1868     \@lnumwidth #3\relax
1869     \advance\leftskip \@lnumwidth \null\nobreak\hskip -\leftskip
1870     {#4}\nobreak
1871     \jsTocLine \nobreak\hb@xt@\@pnumwidth{%
1872       \hfil\normalfont \normalcolor #5}\par}\fi}
```

■ 図目次と表目次

`\listoffigures` 図目次を出力します。

```
1873 \newcommand{\listoffigures}{%
1874 %<*book|report>
1875 \if@twocolumn\@restonecoltrue\onecolumn
1876 \else\@restonecolfalse\fi
1877 \chapter*{\listfigurename}%
1878 \mkboth{\listfigurename}{}%
1879 %</book|report>
1880 %<!*book&!report>
1881 \section*{\listfigurename}%
1882 \mkboth{\listfigurename}{\listfigurename}%
1883 %</!*book&!report>
1884 \starttoc{lof}%
1885 %<book|report> \if@restonecol\twocolumn\fi
1886 }
```

`\l@figure` 図目次の項目を出力します。

```
1887 \newcommand*\l@figure{\dottedtocline{1}{1\zw}{3.683\zw}}
```

`\listoftables` 表目次を出力します。

```
1888 \newcommand{\listoftables}{%
1889 %<*book|report>
1890 \if@twocolumn\@restonecoltrue\onecolumn
```

```

1891 \else\@restonecolfalse\fi
1892 \chapter*{\listtablename}%
1893 \@mkboth{\listtablename}{}%
1894 %</book|report>
1895 %<!*book&!report>
1896 \section*{\listtablename}%
1897 \@mkboth{\listtablename}{\listtablename}%
1898 %</!book&!report>
1899 \@starttoc{lot}%
1900 %<book|report> \if@restonecol\twocolumn\fi
1901 }

```

`\l@table` 表目次は図目次と同じです。

```
1902 \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。

```

1903 \newdimen\bibindent
1904 \setlength\bibindent{2\zw}

```

`thebibliography` 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。

```

1905 \newenvironment{thebibliography}[1]{%
1906 \global\@jssc@warnoldfontcmdexceptiontrue
1907 \global\let\presectionname\relax
1908 \global\let\postsectionname\relax
1909 %<article|jspf> \section*{\refname}\@mkboth{\refname}{\refname}%
1910 %<*kiyou>
1911 \vspace{1.5\baselineskip}
1912 \subsubsection*{\refname}\@mkboth{\refname}{\refname}%
1913 \vspace{0.5\baselineskip}
1914 %</kiyou>
1915 %<book|report> \chapter*{\bibname}\@mkboth{\bibname}{}%
1916 %<book|report> \addcontentsline{toc}{chapter}{\bibname}%
1917 \list{\@biblabel{\@arabic\c@enumiv}}%
1918     {\settowidth\labelwidth{\@biblabel{#1}}%
1919     \leftmargin\labelwidth
1920     \advance\leftmargin\labelsep
1921     \@openbib@code
1922     \usecounter{enumiv}%
1923     \let\p@enumiv\@empty
1924     \renewcommand\theenumiv{\@arabic\c@enumiv}}%
1925 %<kiyou> \small
1926 \sloppy

```

```

1927 \clubpenalty4000
1928 \@clubpenalty\clubpenalty
1929 \widowpenalty4000%
1930 \sfcode`.\@m}
1931 {\def\@noitemerr
1932   {\@latex@warning{Empty `thebibliography' environment}}%
1933 \endlist
1934 \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。

```
1935 \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。

```
1936 \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。

```
1937 % \def\@biblabel#1{\inhibitglue [#1] \inhibitglue}
```

`\cite` 文献の番号を出力する部分は `ltbibl.dtx` で定義されていますが、コンマとカッコを和文
`\@cite` フォントにするには次のようにします。とりあえずコメントアウトしておきましたので、必
`\@citex` 要に応じて生かしてください。かっこの前後に入るグルーを `\inhibitglue` で取っ
ておきますので、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。

```

1938 % \def\@citex[#1]#2{\leavevmode
1939 %   \let\@citea\@empty
1940 %   \@cite{\@for\@citeb:=#2\do
1941 %     {\@citea\def\@citea{, \inhibitglue\penalty\@m\ }%
1942 %     \edef\@citeb{\expandafter\@firstofone\@citeb\@empty}}%
1943 %     \if@filesw\immediate\write\@auxout{\string\citation{\@citeb}}\fi
1944 %     \@ifundefined{b@\@citeb}{\mbox{\normalfont\bfseries ?}}%
1945 %     \G@refundefinedtrue
1946 %     \@latex@warning
1947 %       {Citation `@\@citeb' on page \thepage \space undefined}}%
1948 %     {\@cite@ofmt{\csname b@\@citeb\endcsname}}}{#1}}
1949 % \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。

```

1950 % \DeclareRobustCommand\cite{\unskip
1951 %   \@ifnextchar [{\@tempswatrue\@citex}{\@tempswafalse\@citex []}]
1952 % \def\@cite#1#2{$^{\hbox{\scriptsize{#1}\if@tempswa
1953 %   , \inhibitglue\ #2\fi) }}$}

```

11.3 索引

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

```
1954 \newenvironment{theindex}{% 索引を 3 段組で出力する環境
1955     \if@twocolumn
1956         \onecolumn\@restonecolfalse
1957     \else
1958         \clearpage\@restonecoltrue
1959     \fi
1960     \columnseprule.4pt \columnsep 2\zw
1961     \ifx\multicols\@undefined
1962 %<book|report>     \twocolumn[\@makeschapterhead{\indexname}]%
1963 %<book|report>     \addcontentsline{toc}{chapter}{\indexname}]%
1964 %<!book&!report>     \def\presectionname{}\def\postsectionname{}%
1965 %<!book&!report>     \twocolumn[\section*{\indexname}]%
1966     \else
1967         \ifdim\textwidth<\fullwidth
1968             \setlength{\evensidemargin}{\oddsidemargin}
1969             \setlength{\textwidth}{\fullwidth}
1970             \setlength{\linewidth}{\fullwidth}
1971 %<book|report>     \begin{multicols}{3}[\chapter*{\indexname}]%
1972 %<book|report>     \addcontentsline{toc}{chapter}{\indexname}]%
1973 %<!book&!report>     \def\presectionname{}\def\postsectionname{}%
1974 %<!book&!report>     \begin{multicols}{3}[\section*{\indexname}]%
1975     \else
1976 %<book|report>     \begin{multicols}{2}[\chapter*{\indexname}]%
1977 %<book|report>     \addcontentsline{toc}{chapter}{\indexname}]%
1978 %<!book&!report>     \def\presectionname{}\def\postsectionname{}%
1979 %<!book&!report>     \begin{multicols}{2}[\section*{\indexname}]%
1980     \fi
1981     \fi
1982 %<book|report>     \@mkboth{\indexname}{}%
1983 %<!book&!report>     \@mkboth{\indexname}{\indexname}%
1984     \plainifnotempty % \thispagestyle{plain}
1985     \parindent\z@
1986     \parskip\z@ \@plus .3\jcs@mp@relax
1987     \let\item\@idxitem
1988     \raggedright
1989     \footnotesize\narrowbaselines
1990 }{
1991     \ifx\multicols\@undefined
1992         \if@restonecol\onecolumn\fi
1993     \else
1994         \end{multicols}
1995     \fi
1996     \clearpage
```

1997 }

- `\@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` 1998 `\newcommand{\@idxitem}{\par\hangindent 4\zw} % 元 40pt`
- `\subsubitem` 1999 `\newcommand{\subitem}{\@idxitem \hspace*{2\zw}} % 元 20pt`
2000 `\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。
2001 `\newcommand{\indexspace}{\par \vskip 10\jsc@empt \@plus5\jsc@empt \@minus3\jsc@empt\relax}`
- `\seenname` 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。
2002 `\newcommand\seenname{\if@english see\else →\fi}`
2003 `\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。
2004 `\let\footnotes@ve=\footnote`
2005 `\def\footnote{\inhibitglue\footnotes@ve}`
2006 `\let\footnotemarks@ve=\footnotemark`
2007 `\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のを、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 `newttext` や `newpertext` の使用時におかしくなってしまう。これらのパッケージは内部で `\thefootnote` を再定義していますので、気になる

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

```
2008 \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。

```
2009 % \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。

```
2010 \renewcommand\footnoterule{%
2011   \kern-3\jsc@empt
2012   \hrule width .4\columnwidth height 0.4\jsc@empt
2013   \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。

```
2014 %<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。

```
2015 \long\def\@footnotetext{%
2016   \insert\footins\bgroup
2017     \normalfont\footnotesize
2018     \interlinepenalty\interfootnotelinepenalty
2019     \splittopskip\footnotesepp
2020     \splitmaxdepth \dp\strutbox \floatingpenalty \@MM
2021     \hsize\columnwidth \@parboxrestore
2022     \protected@edef\@currentlabel{%
2023       \csname p@footnote\endcsname\@thefnmark
2024     }%
2025     \color@begingroup
2026     \@makefntext{%
2027       \rule\z@\footnotesepp\ignorespaces}%
2028     \futurelet\jsc@next\jsc@fo@t}
2029 \def\jsc@fo@t{\ifcat\bgroup\noexpand\jsc@next \let\jsc@next\jsc@fo@t
2030               \else \let\jsc@next\jsc@fo@t\fi \jsc@next}
2031 \def\jsc@fo@t{\bgroup\aftergroup\jsc@@foot\let\jsc@next}
2032 \def\jsc@fo@t#1{#1\jsc@@foot}
2033 \def\jsc@@foot{\@finalstrut\strutbox\color@endgroup\egroup}
```

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

```
2034 \newcommand\@makefntext[1]{%
2035   \advance\leftskip 3\zw
2036   \parindent 1\zw
```



```
2037 \noindent
2038 \llap{\@makefnmark\hskip0.3\zw}#1}
```

`\@footnotenext` 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。

```
2039 % \def\@footnotenext[#1]{%
2040 %   \begingroup
2041 %     \ifnum#1>\z@
2042 %       \csname c@\@mpfn\endcsname #1\relax
2043 %       \unrestored@protected@xdef\@thefnmark{\thempfn}%
2044 %     \else
2045 %       \unrestored@protected@xdef\@thefnmark{}%
2046 %     \fi
2047 %   \endgroup
2048 %   \@footnotetext}
```

12 段落の頭へのグルー挿入禁止

段落頭のかぎかっこなどを見かけ 1 字半下げから全角 1 字下げに直します。

[2012-04-24 LTJ] LuaTeX-ja 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、LuaTeX-ja 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。

```
2049 \let\@inhibitglue=\ltjfakeparbegin
2050 \def\@item[#1]{%
2051   \if@noperitem
2052     \@donoperitem
2053   \else
2054     \if@inlabel
2055       \indent \par
2056     \fi
2057     \ifhmode
2058       \unskip\unskip \par
2059     \fi
2060     \if@newlist
2061       \if@nobreak
2062         \@nbitem
2063       \else
2064         \addpenalty\@beginparpenalty
```

```

2065     \addvspace\@topsep
2066     \addvspace{-\parskip}%
2067     \fi
2068   \else
2069     \addpenalty\@itempenalty
2070     \addvspace\itemsep
2071     \fi
2072   \global\@inlabeltrue
2073 \fi
2074 \everypar{%
2075   \@minipagefalse
2076   \global\@newlistfalse
2077   \if@inlabel
2078     \global\@inlabelfalse
2079     {\setbox\z@\lastbox
2080      \ifvoid\z@
2081        \kern-\itemindent
2082        \fi}%
2083     \box\@labels
2084     \penalty\z@
2085   \fi
2086   \if@nobreak
2087     \@nobreakfalse
2088     \clubpenalty \@M
2089   \else
2090     \clubpenalty \@clubpenalty
2091     \everypar{}}%
2092 \fi\ltjfakeparbegin}%
2093 \if@noitemarg
2094   \@noitemargfalse
2095   \if@nmbbrlist
2096     \refstepcounter\@listctr
2097   \fi
2098 \fi
2099 \sbox\@tempboxa{\makelabel{#1}}%
2100 \global\setbox\@labels\hbox{%
2101   \unhbox\@labels
2102   \hskip \itemindent
2103   \hskip -\labelwidth
2104   \hskip -\labelsep
2105   \ifdim \wd\@tempboxa >\labelwidth
2106     \box\@tempboxa
2107   \else
2108     \hbox to\labelwidth {\unhbox\@tempboxa}%
2109   \fi
2110   \hskip \labelsep}%
2111 \ignorespaces}

```

`\@gnewline` についてはちょっと複雑な心境です。もともとの p_{La}T_EX 2_ε は段落の頭にグ

ルーが入る方で統一されていました。しかし \\ の直後にはグルーが入らず、不統一でした。そこで \\ の直後にもグルーを入れるように直していただいた経緯があります。しかし、ここでは逆にグルーを入れない方で統一したいので、また元に戻してしまいました。

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

[2016-12-05 LTJ] 本家 [2016-11-29], lljcore.sty での変更に従います。

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

```
2112 \def\@gnewline #1{%
2113   \ifvmode
2114     \@nolnerr
2115   \else
2116     \unskip \reserved@e {\reserved@f#1}\nobreak \hfil \break \null
2117     \inhibitglue \ignorespaces
2118   \fi}
```

13 いろいろなロゴ

L^AT_EX 関連のロゴを作り直します。

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

[2016-07-21 LTJ] jsclasses と Lua_T_EX-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。

```
2119 \IfFileExists{jslogo.sty}{\@jslogofalse}%
2120 \if@jslogo
2121   \RequirePackage{jslogo}
2122   \def\小{\jsgl@small}
2123   \def\上小{\jsgl@uppersmall}
2124 \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 2125 \def\小#1{\hbox{${m@th}$%
2126   \csname S@f@size\endcsname
2127   \fontsize\sf@size\z@
2128   \math@fontsfalse\selectfont
2129   #1}}
2130 \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)。

```
2131 \def\cmrTeX{%
2132   \ifdim \fontdimen\@ne\font >\z@
2133     T\kern-.25em\lower.5ex\hbox{E}\kern-.125em\@
```

```

2134 \else
2135   T\kern-.1667em\lower.5ex\hbox{E}\kern-.125emX\@
2136 \fi}
2137 \def\cmrLaTeX{%
2138 \ifdim \fontdimen\@ne\font >\z@
2139   L\kern-.32em\上小{A}\kern-.22em\cmrTeX
2140 \else
2141   L\kern-.36em\上小{A}\kern-.15em\cmrTeX
2142 \fi}
2143 \def\sfTeX{T\kern-.1em\lower.4ex\hbox{E}\kern-.07emX\@}
2144 \def\sfLaTeX{L\kern-.25em\上小{A}\kern-.08em\sfTeX}
2145 \def\ptmTeX{%
2146 \ifdim \fontdimen\@ne\font >\z@
2147   T\kern-.12em\lower.37ex\hbox{E}\kern-.02emX\@
2148 \else
2149   T\kern-.07em\lower.37ex\hbox{E}\kern-.05emX\@
2150 \fi}
2151 \def\ptmLaTeX{%
2152 \ifdim \fontdimen\@ne\font >\z@
2153   L\kern-.2em\上小{A}\kern-.1em\ptmTeX
2154 \else
2155   L\kern-.3em\上小{A}\kern-.1em\ptmTeX
2156 \fi}
2157 \def\pncTeX{%
2158 \ifdim \fontdimen\@ne\font >\z@
2159   T\kern-.2em\lower.5ex\hbox{E}\kern-.08emX\@
2160 \else
2161   T\kern-.13em\lower.5ex\hbox{E}\kern-.13emX\@
2162 \fi}
2163 \def\pncLaTeX{%
2164 \ifdim \fontdimen\@ne\font >\z@
2165   L\kern-.3em\上小{A}\kern-.1em\pncTeX
2166 \else
2167   L\kern-.3em\上小{A}\kern-.1em\pncTeX
2168 \fi}
2169 \def\pplTeX{%
2170 \ifdim \fontdimen\@ne\font >\z@
2171   T\kern-.17em\lower.32ex\hbox{E}\kern-.15emX\@
2172 \else
2173   T\kern-.12em\lower.34ex\hbox{E}\kern-.1emX\@
2174 \fi}
2175 \def\pplLaTeX{%
2176 \ifdim \fontdimen\@ne\font >\z@
2177   L\kern-.27em\上小{A}\kern-.12em\pplTeX
2178 \else
2179   L\kern-.3em\上小{A}\kern-.15em\pplTeX
2180 \fi}
2181 \def\ugmTeX{%
2182 \ifdim \fontdimen\@ne\font >\z@

```

```

2183     T\kern-.1em\lower.32ex\hbox{E}\kern-.06emX\@
2184   \else
2185     T\kern-.12em\lower.34ex\hbox{E}\kern-.1emX\@
2186   \fi}
2187 \def\ugmLaTeX{%
2188   \ifdim \fontdimen\@ne\font >\z@
2189     L\kern-.2em\上小{A}\kern-.13em\ugmTeX
2190   \else
2191     L\kern-.3em\上小{A}\kern-.13em\ugmTeX
2192   \fi}
2193 \DeclareRobustCommand{\TeX}{%
2194   \def\@tempa{cmr}%
2195   \ifx\f@family\@tempa\cmrTeX
2196   \else
2197     \def\@tempa{ptm}%
2198     \ifx\f@family\@tempa\ptmTeX
2199     \else
2200       \def\@tempa{txr}%
2201       \ifx\f@family\@tempa\ptmTeX
2202       \else
2203         \def\@tempa{pnc}%
2204         \ifx\f@family\@tempa\pncTeX
2205         \else
2206           \def\@tempa{ppl}%
2207           \ifx\f@family\@tempa\pplTeX
2208           \else
2209             \def\@tempa{ugm}%
2210             \ifx\f@family\@tempa\ugmTeX
2211             \else\sfTeX
2212             \fi
2213           \fi
2214         \fi
2215       \fi
2216     \fi
2217   \fi}
2218
2219 \DeclareRobustCommand{\LaTeX}{%
2220   \def\@tempa{cmr}%
2221   \ifx\f@family\@tempa\cmrLaTeX
2222   \else
2223     \def\@tempa{ptm}%
2224     \ifx\f@family\@tempa\ptmLaTeX
2225     \else
2226       \def\@tempa{txr}%
2227       \ifx\f@family\@tempa\ptmLaTeX
2228       \else
2229         \def\@tempa{pnc}%
2230         \ifx\f@family\@tempa\pncLaTeX
2231         \else

```

```

2232         \def\@tempa{ppl}%
2233         \ifx\f@family\@tempa\pplLaTeX
2234         \else
2235         \def\@tempa{ugm}%
2236         \ifx\f@family\@tempa\ugmLaTeX
2237         \else\sfLaTeX
2238         \fi
2239         \fi
2240     \fi
2241 \fi
2242 \fi
2243 \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。

```

2244 \DeclareRobustCommand{\LaTeXe}{\mbox{%
2245 \if b\expandafter\@car\f@series\@nil\boldmath\fi
2246 \LaTeX\kern.15em2\raisebox{-.37ex}{\textstyle\varepsilon}}}%

```

\pTeX pTeX, pL^AT_EX 2_ε のロゴを出す命令です。

```

\pLaTeX 2247 \def\pTeX{p\kern-.05em\TeX}
\pLaTeXe 2248 \def\pLaTeX{p\LaTeX}
2249 \def\pLaTeXe{p\LaTeXe}

```

\AmSTeX `amstex.sty` で定義されています。

```

2250 \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 2251 % \@ifundefined{BibTeX}
2252 %     {\def\BibTeX{{\rmfamily B\kern-.05em%
2253 %     \textsc{i\kern-.025em b}\kern-.08em%
2254 %     T\kern-.1667em\lower.7ex\hbox{E}\kern-.125emX}}}{%
2255 \DeclareRobustCommand{\BibTeX}{B\kern-.05em\small{I\kern-.025em B}}%
2256 \ifx\f@family\cmr\kern-.08em\else\kern-.15em\fi\TeX}
2257 \DeclareRobustCommand{\SliTeX}{%
2258 S\kern-.06emL\kern-.18em\small{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。

```

2259 \fi

```

14 初期設定

■いろいろな語

```

\prepartname
\postpartname 2260 \newcommand{\prepartname}{\if@english Part~\else 第\fi}
\prechaptername 2261 \newcommand{\postpartname}{\if@english\else 部\fi}
\postchaptername
\postsectionname
\presectionname
\postsectionname

```

```

2262 %<book|report>\newcommand{\prechaptername}{\if@english Chapter~\else 第 \fi}
2263 %<book|report>\newcommand{\postchaptername}{\if@english\else 章 \fi}
2264 \newcommand{\presectionname}{}% 第
2265 \newcommand{\postsectionname}{}% 節

\contentsname

\listfigurename 2266 \newcommand{\contentsname}{\if@english Contents\else 目次 \fi}
\listtablename 2267 \newcommand{\listfigurename}{\if@english List of Figures\else 図目次 \fi}
2268 \newcommand{\listtablename}{\if@english List of Tables\else 表目次 \fi}

\refname

\bibName 2269 \newcommand{\refname}{\if@english References\else 参考文献 \fi}
\indexname 2270 \newcommand{\bibname}{\if@english Bibliography\else 参考文献 \fi}
2271 \newcommand{\indexname}{\if@english Index\else 索引 \fi}

\figurename

\tablename 2272 %<!jspf>\newcommand{\figurename}{\if@english Fig.~\else 図 \fi}
2273 %<jspf>\newcommand{\figurename}{Fig.~}
2274 %<!jspf>\newcommand{\tablename}{\if@english Table~\else 表 \fi}
2275 %<jspf>\newcommand{\tablename}{Table~}

\appendixname

\abstractname 2276 % \newcommand{\appendixname}{\if@english Appendix~\else 付録 \fi}
2277 \newcommand{\appendixname}{\if@english \else 付録 \fi}
2278 %<!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 2019 年 8 月 25 日で、和暦では令和元年 8 月 25 日です。

```

\today
2279 \newif\if 西暦 \西暦 true
2280 \def\西暦{\西暦 true}
2281 \def\和暦{\西暦 false}
2282 \newcount\heisei \heisei\year \advance\heisei-1988\relax
2283 \def\pltx@today@year@#1{%
2284   \ifnum\numexpr\year-#1=1 元 \else
2285     \ifnum\ltjgetparameter{direction}=3
2286       \kansuji\numexpr\year-#1\relax
2287     \else
2288       \number\numexpr\year-#1\relax\nobreak
2289     \fi
2290   \fi 年
2291 }
2292 \def\pltx@today@year{%
2293   \ifnum\numexpr\year*10000+\month*100+\day<19890108
2294     昭和 \pltx@today@year@{1925}%
2295   \else\ifnum\numexpr\year*10000+\month*100+\day<20190501

```

```

2296  平成 \pltx@today@year@{1988}%
2297   \else
2298   令和 \pltx@today@year@{2018}%
2299   \fi\fi}
2300 \def\today{%
2301   \if@english
2302     \ifcase\month\or
2303       January\or February\or March\or April\or May\or June\or
2304       July\or August\or September\or October\or November\or December\fi
2305     \space\number\day, \number\year
2306   \else\if 西暦
2307     \ifnum\ltjgetparameter{direction}=3 \kansuji\year
2308     \else\number\year\nobreak\fi 年
2309   \else
2310     \pltx@today@year
2311   \fi
2312   \ifnum\ltjgetparameter{direction}=3
2313     \kansuji\month 月
2314     \kansuji\day 日
2315   \else
2316     \number\month\nobreak 月
2317     \number\day\nobreak 日
2318   \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）

```

2319 \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。stfloats パッケージがシステムにインストールされている場合は、このパッケージを使って pLaTeX 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。

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

```

2320 %<article>\if@slide \pagestyle{empty} \else \pagestyle{plain} \fi
2321 %<book>\if@report \pagestyle{plain} \else \pagestyle{headings} \fi
2322 %<report|kiyou>\pagestyle{plain}
2323 %<jspf>\pagestyle{headings}
2324 \pagenumbering{arabic}
2325 \fnfixbottomtrue % 2017-02-19
2326 \IfFileExists{stfloats.sty}{\RequirePackage{stfloats}\fnbelowfloat}{}
2327 \if@twocolumn
2328   \twocolumn
2329   \sloppy
2330   \flushbottom
2331 \else
2332   \onecolumn
2333   \raggedbottom

```



```
2334 \fi
2335 \if@slide
2336 \renewcommand\kanjifamilydefault{\gtdefault}
2337 \renewcommand\familydefault{\sfdefault}
2338 \raggedright
2339 \ltj@setpar@global
2340 \ltjsetxkanjiskip0.1em\relax
2341 \fi
```

以上です。