

著者のための L^AT_EX 2_ε

© Copyright 1995–2015, L^AT_EX3 Project Team.
All rights reserved.

30 March 2015 ; 日本語訳は 2018-09-04 です.

Contents

1	イントロダクション	2
1.1	L ^A T _E X 2 _ε — 新しい L ^A T _E X の登場	2
1.2	L ^A T _E X3— L ^A T _E X を長い将来まで使うために	3
1.3	大まかな内容	4
1.4	追加の情報	4
2	クラスとパッケージ	5
2.1	クラスとパッケージと何か	5
2.2	クラスとパッケージのオプション	6
2.3	標準のクラス	7
2.4	標準パッケージ	8
2.5	補助してくれるソフトウェア	9
3	コマンド	11
3.1	初期化コマンド	11
3.2	プレアンプルのコマンド	12
3.3	ドキュメントの構造	14
3.4	定義	14
3.5	ボックス	16
3.6	長さを測る	18
3.7	線の終わり	18
3.8	改ページの制御	18
3.9	フロート	19
3.10	フォントの変更：テキスト	19
3.11	フォントエンコーディング：数式	20
3.12	強制数式モード	21
3.13	本文での添字の設定	21
3.14	テキストコマンド：すべてのエンコーディング	22
3.15	テキストコマンド：T1 エンコーディング	24
3.16	ロゴ	25

3.17 描画コマンド	25
3.18 古いコマンド	26
4 L^AT_EX 2.09 ドキュメント	26
4.1 警告	27
4.2 フォント選択時の問題	27
4.3 ネイティブモード	28
5 自分用の変更	29
6 いくつかの問題	29
6.1 新しいエラーメッセージ	30
6.2 古い内部コマンド	32
6.3 古いファイル	32
6.4 さらに助けが必要な場合	33
7 楽しんで!	33

1 イントロダクション

ドキュメント処理システム L^AT_EX の新しい標準である L^AT_EX 2_ε によるこそ。

このドキュメントでは、新しい L^AT_EX の機能の使い方と、あなたが書いた古い L^AT_EX を L^AT_EX 2_ε で再利用する方法を説明します。このドキュメントは新しい機能についての簡単な入門であり、さらに読者は著者として古い L^AT_EX を使ったことがあることを想定しています。このドキュメントは L^AT_EX 2_ε のリファレンスマニュアルでもなければ、完全な紹介を与えているわけでもありません。

これはいままでは歴史的なドキュメントです。なぜなら L^AT_EX 2_ε は 1994 年から存在しています。

1.1 L^AT_EX 2_ε—新しい L^AT_EX の登場 (とはいうものの、今から 10 年以上も昔ですね.)

以前のバージョンの L^AT_EX は L^AT_EX 2.09 として知られていました。長年にわたり、L^AT_EX 用に多くの拡張が開発されました。これはもちろん継続的な人気のおかげですが、不幸な結果ももたらしました。互換性のない L^AT_EX フォーマットが、いろいろな場所で使用されたのです。それらは、‘標準 L^AT_EX 2.09’、*New Font Selection Scheme* (NFSS) で強化された L^AT_EX、S_LT_EX、A_MS-L^AT_EX などです。したがって、さまざまな場所で作成されたドキュメントを処理するために、システム管理者は複数のバージョンの L^AT_EX プログラムを維持管理しなければなりません。さらに、ソースファイルをみても、そのドキュメントが、どのフォーマットの元で書かれたのかが常に明確であるとは限りませんでした。

この不満足な状況に終止符を打つために、L^AT_EX 2_ε が作られました；L^AT_EX 2_ε は、そのようなすべての拡張を単一フォーマットにまとめあげて、L^AT_EX 2_ε 2.09

との互換性のない方言が広まるのを防ぎます。L^AT_EX 2_ε では、‘新しいフォント選択スキーム (new font selection scheme)’ が標準となり、たとえば、`amsmath` (以前は `AMS-LATEX` フォーマット) や `slides` (以前は `SLATEX` フォーマット) は単なる拡張機能となりました。これらは同一の基本フォーマットを使用したドキュメントによって読み込まれます。

新しいリリースの導入により、頻繁に要求されるいくつかの機能を追加すること、パッケージやクラスを書く作業をシンプルにします。

1.2 L^AT_EX3— L^AT_EX を長い将来まで使うために

L^AT_EX 2_ε は、L^AT_EX 2_ε システムの包括的な再実装における統合ステップです。L^AT_EX 2_ε の次の大きなリリースは、L^AT_EX3 になります。これには、ドキュメントデザイナーとパッケージ作者のための、L^AT_EX 2_ε へのインターフェースの抜本的な見直しが含まれます。

L^AT_EX3 は長期的な研究プロジェクトですが、プロジェクトチームは完了するまでは、L^AT_EX 2_ε に対して積極的なメンテナンスを行います。したがって、L^AT_EX 2_ε の提供とメンテナンスから得られた経験は、L^AT_EX3 の設計に大きな影響を与えます。プロジェクトの簡単な説明は、`ltx3info.tex` にあります。

このプロジェクトをサポートしてくださるなら、L^AT_EX3 プロジェクト基金 (L^AT_EX3 Project Fund) に寄付をお寄せください；これは現在の L^AT_EX を維持し、L^AT_EX3 を開発するために必要なさまざまな経費について、研究チームを経済的に支援するために設立されました。

基金は The T_EX Users Group や、それぞれの地域のユーザグループによって管理されます。寄付をされる場合や、ユーザグループに参加するための情報は次のところにあります：

<http://www.tug.org/lugs.html>

L^AT_EX3 プロジェクトのホームページは

<http://www.latex-project.org/>

です。ここには L^AT_EX と L^AT_EX3 についての紹介があります。そして、L^AT_EX についての他の情報源の案内があります。たとえば、ユーザガイド、T_EX で頻繁に尋ねられる質問 (T_EX Frequently Asked Questions)、そして L^AT_EX のデータベースがあります。

古いドキュメントですが L^AT_EX3 についてのドキュメントは Comprehensive T_EX Archive の元で、ディレクトリ

`ctan:info/ltx3pub`

から、匿名 ftp で公開されています。このディレクトリにある `ltx3pub.bib` には、それぞれのファイルの概略が書かれています。

1.3 大まかな内容

このドキュメントには、 \LaTeX の新しい構造と機能の概要が含まれています。これはバージョン 2.09 以降に変更された \LaTeX の機能だけを含んでいるので、自己完結型のドキュメントではありません。読者は \LaTeX の紹介と合わせてこのドキュメントを読むべきです。

セクション 2 ここでは \LaTeX ドキュメントの新しい構造についての大きな説明です。クラスとパッケージがどのように働くのか、そしてそれらのオプションの使い方を説明しています。 \LaTeX と一緒に提供されている標準パッケージとクラスの一覧を示しています。

セクション 3 ここでは $\text{\LaTeX} 2_{\epsilon}$ の著者のための新しいコマンドを説明しています。

セクション 4 ここでは、 $\text{\LaTeX} 2_{\epsilon}$ で古い \LaTeX を処理する方法を示しています。

セクション 6 ここでは $\text{\LaTeX} 2_{\epsilon}$ を使っていると目にする問題についての解決法のアドバイスを説明しています。 $\text{\LaTeX} 2_{\epsilon}$ ユーザにとっては、新しいいくつかのエラーメッセージと、よく現れる問題と、それへの対処方法、あるいは、より詳しい情報が期にあるかを示しています。

1.4 追加の情報

新しい機能を含んだ \LaTeX である $\text{\LaTeX} 2_{\epsilon}$ の一般的な入門として、必ず Leslie Lamport[4] の著書 *\LaTeX : A Document Preparation System* を読むべきです。

\LaTeX について、さらに詳しいことは、200 のパッケージと、すぐに実行できる 1000 の例とともに Frank Mittelbach と Michel Goossens[5] の著書 *The \LaTeX Companion second edition* に書かれています。

グラフィックスの生成と操作についてのパッケージとプログラムについては Michel Goossens と Sebastian Rahtz と Frank Mittelbach の著書 [1] *The \LaTeX Graphics Companion* に詳しく説明されています

\LaTeX で作成したドキュメントを World Wide Web で公開することについては Michel Goossens と Sebastian Rahtz[2] の著書 *The \LaTeX Web Companion* で説明されています。

新しい \LaTeX のパッケージについては、 \LaTeX と一緒に配布されているそれぞれのパッケージに付属するドキュメントを読んでください。

\LaTeX は多数の解説ドキュメントと一緒に配布されます。 \LaTeX は 6 ヶ月ごとにリリースされ、それには *\LaTeX News* が含まれます；内容はファイル `ltnews*.tex` にあります。クラスとパッケージ作成者ためのガイドが *$\text{\LaTeX} 2_{\epsilon}$ for Class and Package Writers* ですが、これには新しい \LaTeX 機能にもとづいたドキュメントクラスとパッケージの作者のための説明です；ファイル `clsguide.tex` にあります。ガイド *$\text{\LaTeX} 2_{\epsilon}$ Font Selection* は、クラスライターとパッケージライターの

ための L^AT_EX フォント選択スキームを説明しています; `fntguide.tex` にあります. L^AT_EX でのキリル語のサポートは, *Cyrillic languages support in L^AT_EX* に記述されています.

ドキュメントのソースコード (`latex.ltx` からカーネル形式を生成するために使用されるファイルから生成されます) は, *The LaTeX Sources* にあります. この非常に大きなドキュメントには, L^AT_EX コマンドのインデックスも含まれています. これは, `base` ディレクトリで L^AT_EX ファイル `source2e.tex` を, ここにあるソースファイルとクラスファイル `ltxdoc.cls` を使用してタイプセットすることで得られます.

T_EX と L^AT_EX についてもっと詳しく知りた人は, お近くの T_EX ユーザーズグループ, または国際的な T_EX Users グループ (3 参照) にお問い合わせください.

2 クラストパッケージ

このセクションでは, L^AT_EX ドキュメントの新しい構造と, *classes* と *packages* という新しいタイプのファイルについて説明します.

2.1 クラスとパッケージと何か

L^AT_EX 2.09 と L^AT_EX 2_ε の主な違いは `\begin{document}` の前に置くコマンドです.

L^AT_EX 2.09 ではドキュメントの種類を決めるために `article` あるいは `book` というスタイルが使われていました. そして `twoside` あるいは `epsfig` のようなオプションを指定していました. これらは `\documentstyle` によって指定されるようになりました.

```
\documentstyle[options]{style}
```

たとえば, 二段組み (`two-sided`) の論文 (`article`) で, ポストスクリプトの図が使われているなら:

```
\documentstyle[twoside,epsfig]{article}
```

のようにします. しかしながら, 2 種類の異なるドキュメントのスタイルオプションがあります: `twoside` のような組み込みのオプションと `epsfig.sty` のようなパッケージを要求するものです. これらは全く異なる目的をもっています. どのような L^AT_EX ドキュメントスタイルでも `epsfig` パッケージを使うことがありますが, 二段組と宣言されたドキュメントは, そのドキュメントスタイルとなります.

このような混乱を避けるために L^AT_EX 2_ε では, 組み込みのオプションとパッケージとを区別します. 新しく `\documentclass` と `\usepackage` コマンドを用い

```
\documentclass[\langle options \rangle]{\langle class \rangle}
\usepackage[\langle options \rangle]{\langle packages \rangle}
```

のようにします。たとえば、二段組の論文でポストスクリプト (eps) の図を用いるなら

```
\documentclass[twoside]{article}
\usepackage{epsfig}
```

のようにします。1行の`\usepackage` コマンドで複数のパッケージの読み込みを指定できます。つまり

```
\usepackage{epsfig}
\usepackage{multicol}
```

のように書く代わりに、まとめて

```
\usepackage{epsfig,multicol}
```

のようにすることもできます。新しい $\text{\LaTeX} 2_{\epsilon}$ は古い $\text{\LaTeX} 2.09$ の`\documentstyle`が処理できるようになっています。このとき $\text{\LaTeX} 2_{\epsilon}$ は $\text{\LaTeX} 2.09$ 互換性モードにはいります。これについては4節で説明しています。

しかし、新しく描くドキュメントに`\documentstyle` コマンドを使ってはいけません。互換性モードの処理は遅く、 $\text{\LaTeX} 2_{\epsilon}$ の新しい特徴を使うことができません。

クラスとパッケージとの違いですが、ドキュメントクラスは拡張子が`.sty`ではなくて`.cls`です。パッケージの拡張子は`.sty`のまま、変更されていません。したがって、ほとんどの $\text{\LaTeX} 2.09$ のパッケージは、そのまま $\text{\LaTeX} 2_{\epsilon}$ でも動作します。

2.2 クラスとパッケージのオプション

$\text{\LaTeX} 2.09$ では、ドキュメントスタイルだけが`twoside`や`draft`のようなオプションをもつことができます。 $\text{\LaTeX} 2_{\epsilon}$ では、クラスもパッケージもオプションをもつことができます。たとえば、二段組でグラフィックスのある論文の場合は、`dvips`ドライバに対して

```
\documentclass[twoside]{article}
\usepackage[dvips]{graphics}
```

のように指示します。

パッケージは共通のオプションを併用できます。たとえば、`color`パッケージを用いるならば

```
\documentclass[twoside]{article}
\usepackage[dvips]{graphics}
\usepackage[dvips]{color}
```

のようにします。

`\usepackage` は複数のオプションを与えることができるので、

```
\documentclass[twoside]{article}
\usepackage[dvips]{graphics,color}
```

のように短く書くこともできます。

さらに、パッケージは、それぞれのオプションが適切なら `\documentclass` コマンドに与えることもできるので、

```
\documentclass[twoside,dvips]{article}
\usepackage{graphics,color}
```

のように書くこともできます。

クラスとパッケージのオプションについて詳しいことは、*The L^AT_EX Companion* と *L^AT_EX 2_ε for Class and Package Writers* で説明されています。

2.3 標準のクラス

次に示すクラスは L^AT_EX と一緒に配布されています。

article 論文用の `article` については *L^AT_EX: A Document Preparation System* を参照してください。

book 書籍用の `book` については *L^AT_EX: A Document Preparation System* を参照してください。

report レポート用の `report` については *L^AT_EX: A Document Preparation System* を参照してください。

letter 手紙用の `letter` については *L^AT_EX: A Document Preparation System* を参照してください。

slides スライド用の `slides` については *L^AT_EX: A Document Preparation System* を参照してください。以前は SLI_TE_X と呼ばれていたものです。

proc これは論文集のためのもので、`article` を基にして作られました。以前は `proc` パッケージでした。

ltxdoc L^AT_EX を説明するドキュメント作成のためのドキュメントクラスです。`article` を基にして作られています。

ltxguide *LaTeX 2_ε for Authors* と *LaTeX 2_ε for Class and Package Writers* のためのドキュメントクラスです。 **article** を基にして作られています。いま読んでいるドキュメントは、この **ltxguide** を使って書かれました。このクラスのレイアウトは将来の **LaTeX** がリリースされた時には変更されるでしょう。

ltnews *LaTeX News* に使われているドキュメントクラスで、 **article** が基になっています。このクラスのレイアウトは将来の **LaTeX** がリリースされた時には変更されるでしょう。

minimal このクラスは、 **LaTeX** クラスファイルで必要とされる最低限 (3 行) のものです。テキストの幅と高さを設定し、 `\normalsize` を定義するだけです。これは主に、 **article** のような ‘完全’ クラスをロードする必要がない状況で、 **LaTeX** コードのデバッグとテストのためにあります。さらに **article** クラスで提供される構造と大きく異なる構造をもつドキュメントのために新しいクラスを、 **article** の影響なく設計したいときには、これを基にして必要な構造を追加すればよいでしょう。 New feature
1995/12/01

2.4 標準パッケージ

次のパッケージは **LaTeX** と一緒に配布されています。

alltt このパッケージは **alltt** 環境を与えます。これは **verbatim** 環境と似ていますが、 `\`, `{}` は、本来の意味をもちます。詳しくは `alltt.dtx` と *LaTeX: A Document Preparation System* を参照してください。 New feature
1994/12/01

doc これは **LaTeX** プログラムのドキュメントをタイプセットするための基本パッケージです。詳しくは `doc.dtx` と *LaTeX: A Document Preparation System* を参照してください。

exscale これは数式フォントを拡大するためにあります。詳しくは `exscale.dtx` と *LaTeX: A Document Preparation System* を参照してください。

fontenc 特定のフォントエンコーディングを **LaTeX** で使う時に使われます。詳しくは `ltoutenc.dtx` を参照してください。

graphpap このパッケージは `\graphpaper` を定義しています。 **picture** 環境で使われます。 New feature
1994/12/01

ifthen ‘if...ならば...そうでなければ do...’ という形のコマンドを与えます。詳しくは `ifthen.dtx` と *The LaTeX Companion* を参照してください。

inputenc **LaTeX** で、どの文字コードで処理するのかという指示を与えます。詳しくは `inputenc.dtx` を参照してください。 New feature
1994/12/01

latexsym **LaTeX 2_ε** はデフォルトでは **LaTeX** 記号をロードしません。このフォントを使うのなら、 **latexsym** パッケージを使います。詳しくは `latexsym.dtx` と *The LaTeX Companion* を参照してください。また、6 節にも説明があります。

makeidx 索引を生成するためのコマンドを提供します。詳しくは *L^AT_EX: A Document Preparation System* と *The L^AT_EX Companion* を参照してください。

newfont L^AT_EX 2.09 のフォントコマンドを New Font Selection Scheme でエミュレートするために使います。詳しくは *The L^AT_EX Companion* を参照してください。

oldfont L^AT_EX 2.09 のフォントコマンドをエミュレートするために使います。詳しくは *The L^AT_EX Companion* を参照してください。

showidx \index コマンドで拾われている索引項目を、それぞれのページに表示します。詳しくは *L^AT_EX: A Document Preparation System* を参照してください。

syntonly タイプセッティングをせずにドキュメントを処理する時に使われます。詳しくは `syntonly.dtx` と *The L^AT_EX Companion* を参照してください。

tracefnt L^AT_EX でフォントのローディングで、どれくらいの情報が使われているかを管理することができます。詳しくは *The L^AT_EX Companion* を参照してください。

2.5 補助してくれるソフトウェア

次のソフトウェアは、L^AT_EX 2_ε と一緒に同じ配布元から入手できます。L^AT_EX: *A Document Preparation System* に記載されているすべてのファイルを得るには、少なくともグラフィックス `graphics` とツール `tools` を入手する必要があります。amsmath パッケージ (amslatex の一部と以前は amstex と呼ばれていました) と babel については、L^AT_EX: *A Document Preparation System* の C.5.2 項の ‘standard packages’ のリストにも示されています。

New description
1998/12/01

amslatex アメリカ数学会のための高度な数学組版のためのパッケージです。これには amsmath パッケージが含まれています。複雑な数式を組版するための多くのコマンドを提供します。これは、アメリカ数学会によって作成されサポートされており、*The L^AT_EX Companion* を参照してください。

babel このパッケージおよび関連ファイルは、多言語でのタイプセットをサポートします。詳しくは *The L^AT_EX Companion* を参照してください。

cyrillic キリル文字の組版に必要なものすべて (ただしフォントは含まれていません)。

New feature
1998/12/01

graphics これには graphics が含まれています。これはグラフィックスを管理と変換、そして他のソフトウェアで作成されたグラフィックスを取り込みます。color パッケージも含まれており、これはカラーでのタイプセットをサポートしています。これらのパッケージについて詳しくは *The L^AT_EX Companion* を参照してください。

psnfss さまざまな種類のポストスクリプトの Type 1 フォントでタイプセットするために必要なもの（ただしフォントは含まれていません）。

tools L^AT_EX3 開発チームによって作成されたさまざまなパッケージ。

これらのパッケージは、いずれもドキュメントが含まれています。また *The L^AT_EX Companion* と *L^AT_EX: A Document Preparation System* のいずれかにも解説があります。

2.5.1 ツール

ここに示したパッケージも、インストールされているでしょう（ただしシステムによっては、少々名前が異なるかもしれません）。

array これらは `array`, `tabular`, `tabular*`環境に機能を加えて拡張したものです。

calc 長さやカウンタの値を指定する時に適切な代数記号を使うためのものです。 New feature
1996/12/01

dcolumn 表を作成する時に‘小数点’で一揃えをするためのものです。使う時には `array` パッケージが必要です。

delarray 配列の周りに大きな‘区切り記号’が必要な時に使います。使う時には `array` パッケージが必要です。

hhline 表の作成の時に水平の位置の微調整をする時に使います。使う時には `array` パッケージが必要です。

longtable 複数のページにまたがる表を作成します。（これは `array` パッケージが必要ではありませんが、両方をロードされていたら、機能が拡張されている方が使われます）

tabularx `tabularx` 環境を定義します。これは `tabular*`と似ていますが、列と列の幅ではなくて、列の幅を変更し、表の幅に合わせます。

afterpage 現在のページの後ろにテキストを入れます。

bm 太字の数式記号を使います。

enumerate `enumerate` 環境の機能拡張バージョンです。

fontsmpl ‘フォント見本’を作成するためのパッケージとテストファイルです。

ftnright 二段組みの時に脚注を右の段の下に置きます。

indentfirst 節などの最初のパラグラフを字下げします。

layout 現在使用しているドキュメントクラスでのページレイアウトを表示します。

multicol コラムの幅を均等にして、テキストをコラムで作成します。

rawfonts L^AT_EX 2.09 の古い内部フォント名でロードされるフォントを指定します。6.2 節を参照してください。

somedefs パッケージオプションを選択して処理します。(rawfonts パッケージで使用されます)。

showkeys \label, \ref, \cite など指定された‘キー’を印字します。推敲のときに使用すると便利です。

theorem ‘定理のような’環境を便利に定義します。

varioref ページの参照を賢く処理をします。

verbatim verbatim 環境の便利な拡張です。

xr 外部のドキュメントに対して相互参照を行います。

xspace コマンド名の後ろにスペースが発生しないというよくある間違いを避けるための‘スマートスペース’コマンドです。

3 コマンド

この節では L^AT_EX 2_ε で加わった新しいコマンドを解説します。詳しいことは *L^AT_EX: A Document Preparation System* と *The L^AT_EX Companion* で説明されています。

3.1 初期化コマンド

初期化コマンドは \documentclass の前に置かなければなりません。

```
\begin{filecontents} {<file-name>}
<file-contents>
\end{filecontents}
```

filecontents 環境とは、単一のドキュメントファイルにパッケージ、オプション、またはその他のファイルの内容をまとめることを意図しています。ドキュメントファイルが L^AT_EX 2_ε で実行されている場合、この環境の本体は、環境の唯一の引数として名前が与えられたファイルにそのまま (verbatim) (コメント行の前に) 書かれます。ただし、そのファイルがすでに存在する場合は、情報メッセージ以外は何も起こりません。

通常の ASCII テキスト文字 (7ビットの可視テキスト) だけが filecontents 環境に含まれていなければなりません。不可視文字である、タブ文字、フォームフィード、8ビット文字などは、filecontents 環境には含めないでください。

タブとフォームフィードは警告を出します。警告では、これらが、それぞれスペースや空白行に変わることを説明します。8ビット文字は T_EX インストールに依存するので、何が起きるかは一般的には予測できません。

`filecontents` 環境は \LaTeX ファイルをインクルードするために使用されます。他のプレーンテキストファイル（たとえば Encapsulated PostScript ファイルなど）の場合は、コメント行を追加しない `filecontents*` 環境を使わなければなりません。

これらの環境は `\documentclass` の前に置くことしか許されていません、このように制限することで、ドキュメントにバンドルされるどのようなパッケージでも、必要に応じて提供できるのです。

3.2 プリアンブルのコマンド

プリアンブルコマンドの変更は、 $\LaTeX 2\epsilon$ ドキュメントを古いドキュメントとの違いが明確になるように意図的に設計されました。コマンドは `\begin{document}` の前に使わなければなりません。

```
\documentclass [option-list] {class-name} [release-date]
```

$\LaTeX 2.09$ のコマンド `\documentstyle` を置き換えたものです。

ドキュメントには `\documentclass` は一つだけしか使うことができません。そして `filecontents` 環境の後ろに置かなければなりません。また、他のコマンドよりは前に置きます。

`<option-list>` はオプションのリストです。それらは `<class-name>` ファイルで定義されているフォーマット要素を変更するためのものです。そして `\usepackage` コマンドが続きます(続いて説明します)。オプション引数 `<release-date>` によって、クラスファイルの最新のリリース日を指定できます。クラスファイルは `YYYY/MM/DD` の形式で日付を含めなければなりません。クラスファイルのバージョンがこの日付より古い場合は、警告が出されます。

たとえば、二段組みの論文を作成するのに、1994年6月よりも後にリリースされた `article.cls` のバージョンを使いたい場合は

```
\documentclass[twocolumn]{article}[1994/06/01]
```

とします。

```
\documentstyle [option-list] {class-name}
```

このコマンドは降りファイルとの互換性のためにサポートされています。本質的には `\documentclass` と同じですが、 $\LaTeX 2.09$ 互換モードが起動します。また、クラスファイルによって処理されなかった `<option-list>` のオプションは、クラスのロードされた後にパッケージとしてロードされます。詳しいことは4にある $\LaTeX 2.09$ 互換モードを参照してください。

`\usepackage [option-list] {package-name} [release-date]`

利用できる`\usepackage`の数に制限はありません。それぞれのパッケージ (`package-name` で示しました) は、新しい要素 (あるいは`\documentclass` コマンドの `class-name` 引数でロードされたクラスファイル定義の変更) を定義します。パッケージファイルは、このようにドキュメントの範囲を拡張します。

`option-list` 引数はオプションのリストを含むことができます。それらは `package-name` ファイルで定義されていたフォーマット要素を変更できます。

上述したように、`release-date` はパッケージファイルの日付を YYYY/MM/DD というフォーマットでのみ含むことができます。パッケージの古いバージョンが見つかった場合は、警告が出されます。

たとえば、`graphics` を `dvips` ドライバの使用するさいに、1994年6月以降の `graphics.sty` をロードするなら、

```
\usepackage[dvips]{graphics}[1994/06/01]
```

と書きます。

それぞれのパッケージは1回だけロードします。同じパッケージが2回以上要求されても、最初の`\usepackage`でオプションの追加の設定がなければ、2回目以降は何もしません。追加されたオプションではエラーメッセージが表示されることがあります。この場合、解決するためには6節を参照してください。

`\usepackage` コマンドの `option-list` で与えられたオプションを処理するだけでなく、それぞれのパッケージは`\documentclass` コマンドの `option-list` も処理します。言い換えると、すべてのパッケージ (正確には動作を指定するすべてのパッケージ) で処理する必要のあるオプションは、それを必要とするパッケージごとに繰り返し指定するのではなく、`\documentclass` コマンドで一度だけ指定するということです。

`\listfiles`

このコマンドがプリアンブルに置かれている場合、(ドキュメントの処理の結果として) 読み込まれたファイルのリストが、実行の最後に端末 (およびログファイル) に表示されます。可能であれば、短い説明も作成されます。

警告: このコマンドは、`\input{file}`あるいは`\include{file}`などの \LaTeX コマンドを使用して読み込まれたファイルのみを一覧表示します。ファイルがプリミティブな \TeX のシンタックス`\input file`を使用して読み込まれた場合は (ファイル名の前後に中括弧 `{ }` は付けられません)、リストには表示されません。 \LaTeX 形式で中括弧を使用しないと、大きな問題が発生し、重要なファイルが上書きされる可能性があります。それを避けるために、常に中括弧に入れます。

New description
1995/12/01

`\setcounter{errorcontextlines}{num}`

\TeX 3で新しいプリミティブ`\errorcontextlines`を導入しました。これはエラーメッセージのフォーマットを管理します。 \LaTeX 2 ϵ は標準の`\setcounter`コ

マンドを通して、`\errorcontextlines` とのインターフェースを提供します。ほとんどの \LaTeX ユーザは、エラーが出たからといって \LaTeX コマンドの内部の定義を見ようとは思わないでしょう。ですから $\text{\LaTeX} 2_{\epsilon}$ はデフォルトで -1 にしています。

3.3 ドキュメントの構造

ドキュメントクラスの `book` には、ドキュメントの構造のための新しいコマンド

```
\frontmatter
\mainmatter
\backmatter
```

が導入されました。

これらのコマンドは、それぞれ `\frontmatter` で前付け（扉、目次、序文）、`\mainmatter` で本文、`\backmatter` で後付け（参考文献、索引、奥付）であることを明確に示します。

3.4 定義

\LaTeX では、コマンドには必須の引数とオプションの引数の両方をもつことができます。たとえば

```
\documentclass[11pt]{article}
```

では、`11pt` という引数がオプションですが、`article` クラスという名前は必須です。

$\text{\LaTeX} 2.09$ ユーザは引数をもつコマンドを定義していましたが、それらは必須の引数でなければなりません。 $\text{\LaTeX} 2_{\epsilon}$ ユーザの場合は、コマンドと環境の両方でオプションの引数を定義できます。

```
\newcommand {<cmd>} [<num>] [<default>] {<definition>}
\newcommand* {<cmd>} [<num>] [<default>] {<definition>}
\renewcommand {<cmd>} [<num>] [<default>] {<definition>}
\renewcommand* {<cmd>} [<num>] [<default>] {<definition>}
```

これらのコマンドには、新しい、つまり 2 番目のオプション引数があります。これは 1 つのオプションの引数を取るコマンドを定義するために使用されます。この新しい引数について、単純な（したがってそれほど実用的ではない）例題によって説明します。

```
\newcommand{\example}[2][YYY]{Mandatory arg: #2;
Optional arg: #1.}
```

これは`\example` を, $\langle definition \rangle$ の中で`#1` と`#2` で示される 2 つの引数をもつコマンドとして定義します. これは珍しいことではありません. しかし, この`\newcommand` (`[YYY]`) に第 2 のオプションの引数を追加することによって, 新しく定義されたコマンド`\example` の最初の引数 (`#1`) は省略可能になり, そのデフォルト値は `YYY` になります.

つまり`\example` の使用結果は

```
\example{BBB}
```

とすれば,

Mandatory arg: BBB; Optional arg: YYY.

とプリントされるか, あるは,

```
\example[XXX]{AAA}
```

とすれば,

Mandatory arg: AAA; Optional arg: XXX.

とプリントされます.

オプション引数のデフォルトの値は `YYY` です. この値は`\example` を作る`\newcommand` の $\langle default \rangle$ 引数で決まります.

別の, もっと便利な例は, 次の定義

```
\newcommand{\seq}[2][n]{\lbrace #2_{0}, \ldots, \, #2_{#1} \rbrace}
```

です. この入力 $\$ \backslash seq \{a\} \$$ は数式 $\{a_0, \dots, a_n\}$ が生成されます. 一方で, 入力 $\$ \backslash seq [k] \{x\} \$$ では $\{x_0, \dots, x_k\}$ が生成されます.

まとめますと, コマンド

```
\newcommand \langle cmd \rangle [ \langle num \rangle ] [ \langle default \rangle ] { \langle definition \rangle }
```

は, $\langle cmd \rangle$ を引数 $\langle num \rangle$ をもつコマンドとして定義します. 最初の引数はオプションで, デフォルトの値は $\langle default \rangle$ です.

注意しなければならないのは, この例はオプション引数が 1 つだけであることです. 以前と同じように, 引数は 9 個までもつことができます.

```

\newenvironment {<cmd>} [<num>] [<default>] {<beg-def>} {<end-def>}
\newenvironment* {<cmd>} [<num>] [<default>] {<beg-def>} {<end-def>}
\renewenvironment {<cmd>} [<num>] [<default>] {<beg-def>} {<end-def>}
\renewenvironment* {<cmd>} [<num>] [<default>] {<beg-def>} {<end-def>}

```

LaTeX 2_ε は、1つのオプション引数を、もつ環境を作ることもサポートしています。2つのコマンドのシンタックスは、`\newcommand` と同じ方針で拡張されています。

```

\providecommand {<cmd>} [<num>] [<default>] {<definition>}
\providecommand* {<cmd>} [<num>] [<default>] {<definition>}

```

これは `\newcommand` と同じ引数を取ります。 `<cmd>` が定義済みならば、存在している定義はそのまま使われます。しかし未定義なら、`\providecommand` の効果は、`\newcommand` で行ったように、 `<cmd>` を定義しておかなければなりません。

上記の5つの‘定義コマンド’はすべて、テキストの全段落を含むことが意図されていない限り、通常、引数付きのコマンドを定義するときに使用される形式である星印*形式をもちます。さらに、星印のない形式を使用する必要がある場合は、その引数が適切に定義された環境の内容として扱われるのが良いかどうかを調べる必要があります。

New feature
1994/12/01

上記で示した5つの‘定義コマンド’によって生成されたコマンドは、今では堅牢 (robust) になりました。

New feature
1995/12/01

3.5 ボックス

次に示す3つのコマンドは LR-boxes 作成のためのもので LaTeX 2.09 に存在していました。これらは、次の方法で拡張されました。

```

\makebox [<width>] [<pos>] {<text>}
\framebox [<width>] [<pos>] {<text>}
\savebox {<cmd>} [<width>] [<pos>] {<text>}

```

LaTeX 2_ε での小さくみえて、とても大きな変化の1つは、 `<width>` 引数のみだけで、4つの特別な長さを使用できることです。これらはボックスの次元をもち、単に `\mbox{<text>}` を使って生成されます。


```

\height  its height above the baseline;
\depth   its depth below the baseline;
\totalheight  the sum of \height and \depth;
\width   its width.

```

したがって、自然な長さ2つ分のボックスの中心に ‘hello’ を置きたい場合は、


```
\makebox[2\width]{hello}
```

と書きたいでしょう。あるいは、四角形のボックスの中に f を 

```
\framebox{\makebox[\totalheight]{\itshape f\}}
```

のようにして入れることもできます。注意しなければならないのは、枠付きのボックスの枠も含めた全部の長さは、`\totalheight` に設定されていることです。

もう1つの変更は、 $\langle pos \rangle$ の新しい可能性です。l と r に加えて s が追加されました。 $\langle pos \rangle$ が s ならば、テキストはボックスの長さにあわせて引き伸ばされ、ボックスの中身には任意の‘ゴム長さ (rubber lengths)’ (ここで単語の間のスペースを含みます) が使用されます。そのような‘ゴム長さ’が与えられていない場合は‘underfull box’ という表示が出ます。

```
\parbox [\langle pos \rangle] [\langle height \rangle] [\langle inner-pos \rangle] {\langle width \rangle} {\langle text \rangle}
\begin{minipage} [\langle pos \rangle] [\langle height \rangle] [\langle inner-pos \rangle] {\langle width \rangle}
\langle text \rangle
\end{minipage}
```

上で示したボックスコマンドと同じような、`\height`、`\width` などがボックスの自然な長さを表すために $[\langle height \rangle]$ 引数で使われます。

L^AT_EX 2_ε では $\langle inner-pos \rangle$ 引数が新しくなりました。`\makebox` などの $\langle pos \rangle$ 引数に垂直で、ボックス内の $\langle text \rangle$ の位置を決定します。 $\langle inner-pos \rangle$ は、t, b, c, あるいは s のいずれか1つであり、それぞれ上, 下, 中央, ‘引伸’して一揃えをします。 $\langle inner-pos \rangle$ 引数が指定されていない場合、L^AT_EX は $\langle pos \rangle$ と同じ値を与えます (これは後者のデフォルト値です)。

```
\begin{lrbox} {\langle cmd \rangle}
\langle text \rangle
\end{lrbox}
```

これは何も直接プリントしない環境の一つです。その効果は、mtext のタイプセットを $\langle cmd \rangle$ に保存することです。したがって、`\sbox {\langle cmd \rangle} {\langle text \rangle}` のようなものですが、 $\langle text \rangle$ の中身の前後の空白は無視されます。

これは、 $\langle text \rangle$ の中で `\verb` コマンドと `verbatim` 環境を使うことができるので便利です。

これは、たとえば‘framed box’環境を定義することにも使うことができます。最初にこの環境を使って $\langle cmd \rangle$ にテキストを保存して、そしてから `\fbox{\usebox{\langle cmd \rangle}}` を呼び出します。

次の例では、`fmpage` を呼び出して環境を定義しています。これは `minipage` の枠付きバージョンです。

```

\newsavebox{\fmbbox}
\newenvironment{fmpage}[1]
  {\begin{lrbox}{\fmbbox}\begin{minipage}{#1}}
  {\end{minipage}\end{lrbox}\fbox{\usebox{\fmbbox}}}

```

3.6 長さを測る

最初のコマンドは L^AT_EX 2.09 にあったものです。新しい 2 つのコマンドが何かは明らかでしょう。

```

\settowidth {<length-cmd>} {<lr text>}
\settoheight {<length-cmd>} {<lr text>}
\settodepth {<length-cmd>} {<lr text>}

```

3.7 線の終わり

コマンド `\` は、様々なところで行の終わりを示すの使われますが、節のタイトルに使われる時になどに、堅牢なコマンドになりました。

New description
1994/12/01

どのような種類の線が終わるのかということを知る必要があります。そこで、次の新しいコマンドを導入しました。これらも `\` と同じ引数を取ります。

```

\tabularnewline [ <vertical-space> ]

```

New feature
1994/12/01

使用例の一つとして、`tabular` 環境の最後の列のテキスト `\raggedright` で設定されています。そして、`\tabularnewline` は `tabular` の行の終わりを示すために使用できますが、`\` は列内の段落内のテキスト行の終わりを示します。このコマンドは `array` 環境でも `tabular` と同じように使うことができます。さらに、この環境の拡張版が `array` パッケージと `longtable` パッケージとして、`tools` に含まれています。

3.8 改ページの制御

ときどき必要となることは、ドキュメントの最終バージョンで L^AT_EX が望ましい箇所改ページをするのを ‘助けてあげる’ ことです。L^AT_EX 2.09 にはこのような状況のためのコマンドがありました。それらは `\clearpage`、`\pagebreak` などで。L^AT_EX 2_ε では、これらに加えて、長いページだけでなく短いページを生成できるコマンドがあります。

```

\enlargethispage {<size>}
\enlargethispage* {<size>}

```

これらのコマンドはページの高さ（標準値は `\textheight`）を `<size>` で指定した量だけ増やします。この変更は現在のページだけに反映します。

これを使うと、例えば、ページに収まるように余分な線を引いたり、負の長さにして通常のページより狭くできます。

星印付きも、テキストを1ページに収まるように可能な限り空白行を縮めます。

これらのコマンドは、脚注のテキストの位置を変更しません。したがって、テキストの量（行数）が多いと脚注と重なってしまいます。

New description
1995/12/01

3.9 フロート

新しいコマンド`\suppressfloats`と新しい‘フロート指示’があります。これらは \LaTeX のフロートを配置するアルゴリズムを、さらによく管理するために使われます。

`\suppressfloats [(placement)]`

このコマンドは、それ以上のフローティング環境が現在のページに置かれないようにします。オプション引数`t`あるいは`b`（どちらか一つ）で制限すれば、フロートをページの上または下に置きます。通常ならこのページに配置されるフロートは、次のページに配置されます。

The extra float location specifier: `!`

フロートの位置決めオプションには、`h`、`t`、`b`と`p`から少なくとも一つを指定します。

`!`が与えられていたら、この特別なフロートは次のようなことは無視されます。

- フロートの数についてのすべての制限、
- テキストページのスペースの量に関する明示的な制限は、フロートかテキストで埋められます。

このメカニズムは、このような状況でもページがオーバーフルにならないように努力し、同じタイプのフロートは正しい順序で印刷されるようにします。

これがフロートのページの生成には影響を与えないことに注意してください。

`!`指定子は、この特別なフロートについての`\suppressfloats` コマンドを無視します。

3.10 フォントの変更：テキスト

\LaTeX 2_ε で使用されているフォント選択スキームは、 \LaTeX 2.09 で使用されているフォント選択スキームと異なることが多いです。この節では、新しいコマンドについて簡単に説明します。詳しい説明と実例については *The \LaTeX Companion*

にあります。クラスとパッケージの製作者向けのインターフェースは *L^AT_EX 2_ε Font Selection* で説明されています。

```
\rmfamily
\sffamily
\ttfamily
\mdseries
\bfseries
\upshape
\itshape
\slshape
\scshape
```

これらはフォントコマンドで、**rm**、**bf** などと同じです。違いは、これらのコマンドではフォントの1つの属性だけを変更することです（変更された属性は名前前の一部です）。これによる結果の1つは、たとえば、`\bfseries\itshape` はシリーズの変更と字形の変化の両方を行うので、太字のイタリック体を与えます。

```
\textrm{text}
\textsf{text}
\texttt{text}
\textmd{text}
\textbf{text}
\textup{text}
\textit{text}
\textsl{text}
\textsc{text}
\emph{text}
```

これらは引数を1つだけもつコマンドです。特定のフォントでタイプセットさせるテキストを引数として取ります。必要であれば斜体補正を自動的に挿入します。その結果が気に入らなければ、`\/`でイタリック補正を加えることも、`nocorr`で削除することもできます。`\nocorr` は`{text}`引数の中で最初か最後に置かなければなりません。

3.11 フォントエンコーディング：数式

数学モードで使用されるフォントのほとんどは、明示的に呼び出す必要はありません。使うことができるフォントから文字を使用するために、以下のコマンド類が用意されています。

```
\mathrm {\langle letters \rangle}
\mathnormal {\langle letters \rangle}
\mathcal {\langle letters \rangle}
\mathbf {\langle letters \rangle}
\mathsf {\langle letters \rangle}
\mathtt {\langle letters \rangle}
\mathit {\langle letters \rangle}
```

さらに引数として文字を1つだけとるコマンドがあり、それは特定のフォントでタイプセットします。引数は数式モードで処理されるので、その中のスペースは無視されます。文字、数字、そしてアクセントだけが変更されます。たとえば、 $\mathbf{\tilde{A} \times 1}$ とすれば $\tilde{A} \times 1$ が生成されます。

3.12 強制数式モード

```
\ensuremath {\langle math commands \rangle}
```

LaTeX 2.09 では、数式モードでもテキストモードでも働くコマンドが必要な場合は、次のようにすることでした。

```
\newcommand{\Gp}{\mbox{$G_p$}}
```

残念ですが、`\mbox` は `\Gp` が添え字や分数に現れる時に、正しい大きさになりません。

LaTeX 2_ε では、次のようにします。

```
\newcommand{\Gp}{\ensuremath{G_p}}
```

こうすれば、`\Gp` は、どの文脈でも正しく働きます。

これが意味することは、`\ensuremath` は何もせずに、`\Gp` が数式モードで呼ばれたら単に `G_p` を生成します。しかし `\Gp` がテキストモードで呼ばれたとき、必要ならば数式モードに入り（処理が終わったら出）ます。

3.13 本文での添字の設定

```
\textsuperscript {\langle text \rangle}
```

LaTeX 2.09 では、脚注記号などのテキストで使われる上付き文字は、内部的に数式モードに入り、数字を数式の上付き文字として生成されました。コンピュータモダンフォントが使用されている場合、数学フォントとテキストフォントの数字は同じであるため、通常は正しい見た目となります。しかし、異なるドキュメントフォント（たとえば Times など）を選択すると、結果の見ためはおかしくなります。このため、`\textsuperscript` コマンドが用意されました。これは引数を現在のテキストフォントを使って上付き文字の位置に正しいサイズでタイプセットします。

New feature
1995/06/01

3.14 テキストコマンド：すべてのエンコーディング

L^AT_EX 2_ε と L^AT_EX 2.09 の主な違いの 1 つは、L^AT_EX 2_ε が任意のエンコーディングのフォントを扱うことができることです。(フォントエンコーディングはフォント内の文字の並びです。たとえば、キリル文字フォントはギリシア語フォントとは異なるエンコーディングになっています。)

New description
1994/12/01

英語やドイツ語などのラテン語に使用される 2 つの主要なフォントエンコーディングは、OT1 (Donald Knuth の 7 ビットエンコーディングです。これは、長い間 T_EX で使用されていました) と T1 (8 ビットの ‘Cork’ エンコーディングです)。

L^AT_EX 2.09 は OT1 エンコードのみがサポートされていますが、L^AT_EX 2_ε は OT1 と T1 のどちらもサポートされています。次の節では、T1 でエンコードされたフォントを使用する時に使う新しいコマンドについて説明します。この節では、すべてのエンコーディングで使用できる新しいコマンドについて説明します。

これらのコマンドのほとんどは L^AT_EX 2.09 でも利用可能な文字を提供しています。たとえば、`\textemdash` は ‘エムダッシュ’ を与えますが、これは L^AT_EX 2.09 では `---` のようにタイプしていました。しかし、いくつかのフォント (たとえばギリシア語のフォントなど) では、`---` のためのリガチャがありませんが `\textemdash` と入力すればエムダッシュを生成できます。

`\r{text}`

New feature
1994/12/01

このコマンドは ‘輪っか’ のアクセントを与えます。たとえば、‘`o`’ を生成するには `\r{o}` というようにタイプします。

`\SS`

New feature
1994/12/01

このコマンドはドイツ語の ‘SS’、つまり大文字の ‘B’ を生成します。この文字の綴りは ‘SS’ ではないので、大文字で入力します。

`\textcircled{text}`

New feature
1994/12/01

このコマンドは `\copyright` のような ‘丸囲み文字’ を作るのに使われます。たとえば `\textcircled{a}` とすれば `@` が生成されます。

`\textcompwordmark`

New feature
1994/12/01

このコマンドは通常ならリガチャされる文字を話して生成するために使われます。たとえば、`f\textcompwordmark i` とすれば、‘fi’ となります。‘f’ と ‘i’ は、‘fi’ となりません。これは英語では、滅多に必要なとはなりません (‘shelfful’ が、稀な例です) が、ドイツ語などでは必要となります。

`\textvisiblespace`

New feature
1994/12/01

このコマンドは‘見える空白’文字‘ ’を生成します。これはコンピュータのプログラムリストなどで、たとえば‘type hello_world’のように使われます。

```
\textemdash \textendash \textexclamdown \textquestiondown
\textquotedblleft \textquotedblright \textquoteleft \textquoteright
```

New feature
1994/12/01

これらのコマンドは特殊な文字の生成かりガチャ（合字）させたいときに使われます。

リガチャ	文 字	コマンド
---	—	<code>\textemdash</code>
--	-	<code>\textendash</code>
!‘	¡	<code>\textexclamdown</code>
?‘	¿	<code>\textquestiondown</code>
‘‘	“	<code>\textquotedblleft</code>
’’	”	<code>\textquotedblright</code>
‘	‘	<code>\textquoteleft</code>
’	’	<code>\textquoteright</code>

これらの文字を作る理由は、そのような文字をもっていないエンコーディングでも生成できるようにするためです。

```
\textbullet \textperiodcentered
```

New feature
1994/12/01

これらのコマンドは、これまで数式モードでのみ使われる文字をテキストで使うためです。

数式コマンド	文 字	テキストコマンド
<code>\bullet</code>	•	<code>\textbullet</code>
<code>\cdot</code>	·	<code>\textperiodcentered</code>

```
\textbackslash \textbar \textless \textgreater
```

New feature
1995/12/01

これらのコマンドは、verbatim モードまたは数式モードでのみ使われる ASCII 文字を使うためです。

数式コマンド	文 字	テキストコマンド
<code>\backslash</code>	\	<code>\textbackslash</code>
<code>\mid</code>		<code>\textbar</code>
<	<	<code>\textless</code>
>	>	<code>\textgreater</code>

```
\textasciicircum \textasciitilde
```

New feature
1995/12/01

これらのコマンドは、verbatim モードでのみ使われる ASCII 文字を使うためです。

verbatim テキストコマンド
 ^ \textasciicircum
 ~ \textasciitilde

`\textregistered \texttrademark`

New feature
 1995/12/01

これらのコマンドは‘登録商標’ (R) 記号と‘商標’(TM) 記号を生成します。

3.15 テキストコマンド : T1 エンコーディング

OT1 フォントのエンコーディングは英語のタイプセットには問題ありませんが、他の言語のタイプセットには問題があります。T1 エンコーディングは、余分な文字（たとえば‘eth’と‘thorn’）を提供することによってこれらの問題のいくつかを解決し、アクセント付きの文字を含む単語のハイフネーション（`babel` のようなパッケージを使えばアメリカ英語以外のハイフネーション）も行えます。

New description
 1994/12/01

この節では、T1 フォントをもっている場合に、それを使う方法を説明します。そのためには、‘ec fonts’、あるいは‘`textf psnfss`’で使われているような T1 でエンコードされた PostScript フォントが必要です。これらすべてのフォントは Comprehensive \TeX アーカイブ (CTAN) の匿名 ftp で取り寄せるか、*forall \TeX* 、または *\TeX Live* という CD-ROM から入手できます。(\TeX ユーザーグループから入手できます)。

これらの準備ができれば、T1 フォントは、次のようにすれば選ぶことができます。

`\usepackage[T1]{fontenc}`

こうすれば、この節で説明しているコマンドが利用できます。

注意：このドキュメントは、最新の \LaTeX を実行しているサイトで処理可能にするために、T1 エンコードされたフォントのみ存在する文字は含みません。つまり、このドキュメントでは、これらのグリフがどうであるかを表示できません。それらを見たいなら、`fontsmpl` ドキュメントを \LaTeX で実行してください。プロンプト‘`cmr`’が現れたらフォントのファミリーネームを与えてください。

`\k{text}`

New feature
 1994/12/01

このコマンドはポーランド語などの‘オゴネク’アクセントを生成します。

`\DH \DJ \NG \TH \dh \dj \ng \th`

New feature
 1994/12/01

これらのコマンドは、文字‘eth’、‘dbar’、‘eng’‘thorn’を生成します。

`\guillemotleft \guillemotright \guilsinglleft \guilsinglright
 \quotedblbase \quotesinglbase \textquotedbl`

New feature
 1994/12/01

これらのコマンドは、いろいろな引用符を生成します。大まかに表示すれば、次のようなものです。<a> <a> „a“ ,a‘そして"a"。

したがって T1 エンコードされフォントだけを使っている場合に必要なりガチャが必要となります。 New description
2001/06/01

ギョメ記号\guillemotleft と\guillemotright¹は、<<と>>、そして\quotedblbase は、,, とすれば得られます。

さらに、予想外の結果ですが、OT1 エンコードフォント、<と>は<と>とすれば得られます。

さらに注意することは、単一の文字"は" を生成しませんが、\textquotedbl とすれば得ることができます。

3.16 ログ

```
\LaTeX  
\LaTeXe
```

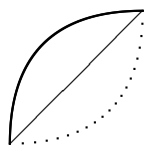
\LaTeX ('L^AT_EX' が生成されます) は、現在でも主なロゴコマンドですが、新しい機能を示したいときには、\LaTeXe ('L^AT_EX 2_ε' が生成されます) とします。

3.17 描画コマンド

```
\qbezier[⟨N⟩](⟨AX⟩,⟨AY⟩)(⟨BX⟩,⟨BY⟩)(⟨CX⟩,⟨CY⟩)  
\bezier{⟨N⟩}(⟨AX⟩,⟨AY⟩)(⟨BX⟩,⟨BY⟩)(⟨CX⟩,⟨CY⟩)
```

\qbezier コマンドは、picture モードで、(⟨AX⟩,⟨AY⟩) から (⟨CX⟩,⟨CY⟩) までの 2 次 Bezier 曲線をコントロール点を (⟨BX⟩,⟨BY⟩) で描くときに使われます。

たとえば、ダイアグラム



は、次のようにして描かれます。

```
\begin{picture}(50,50)  
  \thicklines  
  \qbezier(0,0)(0,50)(50,50)
```

¹私たちは、アドビの多くの特徴をサポートするために、混乱させたことについて、再度謝ります！

```

\qbezier[20](0,0)(50,0)(50,50)
\thinlines
\put(0,0){\line(1,1){50}}
\end{picture}

```

`\bezier` コマンドは、これまでと同じですが、引数 $\langle N \rangle$ はオプションではありません。これは \LaTeX 2.09 の `bezier` ドキュメントのスタイルオプションとの互換性のために与えられています。

3.18 古いコマンド

`\samepage`

`\samepage` コマンドはまだ存在しますが、もはや維持されていません。このコマンドはうまく機能しなかったためです。対象とする範囲内でページ区切りが発生しないことを保証していません。脚注とマージン（余白）が間違った場所に置かれることがあります。

私たちは、これの代わりに `\enlargethispage` をページ区切りコマンド `\newpage` と `\pagebreak` と組み合わせて、ページ区切りを制御することをお勧めします。

`\SLiTeX`

`SLiTeX` は存在しませんので、 \LaTeX ではロゴは定義されていません。代わりに `\textsc{SliTeX}` を使ってください。`SLiTeX` ロゴは、 \LaTeX 2.09 互換モードでは定義されています。

`\mho \Join \Box \Diamond \leadsto`
`\sqsubset \sqsupset \lhd \unlhd \rhd \unrhd`

これらの記号は、 \LaTeX 2.09 で自動的に読み込まれる \LaTeX シンボルフォントには含まれています。しかし、 \TeX は数学フォントファミリーを 16 種しか扱えません。そして多くのユーザーは、それらがなくなったことを発見しました。そのため \LaTeX 2 ϵ は `latexsym` パッケージを使用しない限り、 \LaTeX 記号フォントを読み込みません。

これらの記号は、異なるフォントを使用すれば利用できます。たとえば `amsmath` パッケージには、さらに多くの記号が定義されています。これは、アメリカ数学会から提供されています。

`latexsym` パッケージは、 \LaTeX 2.09 互換モードでは自動的に読み込まれます。

4 \LaTeX 2.09 ドキュメント

\LaTeX 2 ϵ はほとんどの \LaTeX 2.09 ドキュメントを \LaTeX 2.09 互換モードで処理できます。何も変更されていないので、 \LaTeX を同じように実行でき、ほとん

ど同じ結果を得ます。

‘ほとんど’という理由は、いくつかの $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 2.09 パッケージは $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ がサポートしない低レベルの機能を使っているからです。そのようなパッケージを見つけたら、 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 2 ϵ でも機能するようになっているか調べてください。ほとんどのパッケージは、 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 2 ϵ でもそのまま働きます— どのパッケージが問題ないかを確かめるためには、とにかく試してみることです。

$\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 2.09 互換モードは、 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 2.09 をエミュレートしますが、その引き換えに時間がかかります。互換モードは $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 2.09 に比べて 50 パーセントほど遅くなります。その上、 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 2 ϵ の新しい機能は $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 2.09 互換モードにはありません。

4.1 警告

$\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 2.09 互換モードは、2.09 のドキュメントの処理 (かなり昔に書かれたものであってほしいのですが)、つまり古いシステムで書かれたためにその時代の骨董的な $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 2.09 を使いたい、のためだけにあります。

New description
1995/12/01

そのため、このモードは $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 2 ϵ の拡張機能を使うためのものではありません。新しいドキュメントを 2.09 ドキュメントのふりをさせて (つまり `\documentstyle` で始まる) 処理するために使うべきではありません。 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 2 ϵ にだけ備わっているコマンドや環境があるので、骨董的な $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 2.09 システムでは処理できません。

このようなシステムの誤用を防ぐため、そしてまた誤ってエンコードされたドキュメントで引き起こされる問題を回避するために、*LaTeX 2.09* 互換モードは新機能やコマンドのほとんどをオフにします。それらを使用しようとするとエラーメッセージが表示され、さらに多くのものは機能せず、予期しない結果をもたらすことになります。これは意図的にしたことなので、そのようは現象が発生してもバグ報告を私たちに送らないように。

4.2 フォント選択時の問題

互換モードを使用すると、古いドキュメントで行っていたフォント変更コマンドに問題が起こることがあります。これらの問題には 2 種類があります。

- producing error messages;
- not producing the font changes you expected.

このエラーメッセージでは、ドキュメント (あるいは古いスタイルファイルが使われている) は、サポートされなくなった古い内部コマンドへの参照がある場合です。このような場合の情報は 6.2 節にあります。

予期しない結果を得る例の一つは、新しい数式モードのフォント変更コマンドを使った時です。

New description
1995/12/01

```
$ \mathbf{xy} A $
```

次のようにすると

```
$ \bf {xy} A $
```

A も太字になります。

L^AT_EX 2.09 は、L^AT_EX のインストールをカスタマイズすることができたので、インストールされた L^AT_EX が異なれば異なる結果が得られました。L^AT_EX 2_ε は大きなカスタマイズをできないようでしたが、その代わりに古いドキュメントとの互換性のために、ローカルの設定ファイル `latex209.cfg` が L^AT_EX 2_ε が L^AT_EX 2.09 互換モードに入るたびに読み込まれます。

たとえば、使用しているシステムが、新しいフォント選択スキーム (NFSS) を `oldfont` オプションで使用するようになっていれば、L^AT_EX 2_ε のエミュレートはこれに従って `latex209.cfg` ファイルに

```
\ExecuteOptions{oldfont}\RequirePackage{oldfont}
```

が生成されています同じように、`newfont` オプションで NFSS をエミュレートすると、`latex209.cfg` には

```
\ExecuteOptions{newfont}\RequirePackage{newfont}
```

が生成されています。

4.3 ネイティブモード

To run an old document faster, and use the new features of L^AT_EX 2_ε, you should try using *L^AT_EX 2_ε native mode*. This is done by replacing the command: 古いドキュメントの処理を速く処理し、そして L^AT_EX 2_ε の新しい機能を使いたいのなら *L^AT_EX 2_ε ネイティブモード* を使いなさい。コマンドを置き換えるだけです。古い

```
\documentstyle[options],packages]{class}
```

を

```
\documentclass[options]{class}  
\usepackage{latexsym,packages}
```

とします。しかし、いくつかのドキュメントはネイティブモードでは働かず L^AT_EX 2.09 互換モードでしか処理できないでしょう。いくつかの L^AT_EX 2.09 パッケージは LaTeX_ε では L^AT_EX 2.09 互換モードでのみ動作するでしょう。いくつかのドキュメント L^AT_EX 2_ε で改良されたエラー発見機能のために、エラーが出るでしょう。

しかし、ほとんどの L^AT_EX 2.09 ドキュメントは L^AT_EX 2_ε のネイティブモードで、上に示した変更のみで処理できます。もう一度言いますが、あなたのドキュメントがネイティブモードで働くかどうかは、実際に試すことです。

5 自分用の変更

非常に簡単に行うことができる一般的な修正方法が 2 つあります。ただしそのような変更を使用して作成したドキュメントは、他の場所では使用できないことを忘れないでください（このようなドキュメントは‘可搬性がない’と呼ばれます）。

New description
1995/12/01

修正の 1 つのタイプは、頻繁に使用する記号または構造に対して自分用のコマンドの使うことです。これらをパッケージファイル（たとえば、`mymacros.sty` と呼ばれるファイル）に入れて、ドキュメントのプリアンブルで `\usepackage{mymacros}` として読み込みます。

もう一つの方法は、標準クラスに似せて自分用のクラスファイルを作成することです。それには、環境の追加、いくつかのパラメータの値を変更するなど、直接的な変更も含まれます。これらはクラスファイルに入れる必要があります。ここでは、`article` クラスと非常によく似ている `larticle` というクラスを使用して、そのようなファイルを構築する簡単な方法を説明します。

クラスファイル `larticle.cls` は（予備的な識別コマンドの後で）、つぎのように始めなければなりません。

New feature
1995/12/01

```
\LoadClassWithOptions{article}
```

このコマンドに続くのは、`article.sty` ファイルを読み込んだ後で行いたい追加や変更です。

上記の `\LoadClassWithOptions` コマンドを使った効果は、標準クラスファイル `article` をドキュメントで指定されたオプションで読み込むことです。したがって、`larticle` クラスを使用するドキュメントでも標準の `article` クラスを使用するときに指定できるオプションを指定できます。

```
\documentclass[a4paper,twocolumn,dvips]{larticle}
```

6 いくつかの問題

この説では L^AT_EX 2_ε の使い方を間違った時に、どうすればよいかを説明します。

6.1 新しいエラーメッセージ

L^AT_EX 2_ε には新しいエラーメッセージがたくさんあります。これらのエラーメッセージは、エラープロンプトに対して `h` を押せば、多くの助けとなる情報が得られます。

`Option clash for package <package>.`

パッケージは、異なるオプションで 2 回呼び出されました。 `h` を押して、どのオプションを指定したかったのかを与えます。たとえば問題のドキュメントが

```
\usepackage[foo]{fred}
\usepackage[baz]{fred}
```

であるとすれば、エラーメッセージは

```
Option clash for package fred.
```

となるでしょう。ここで `?` プロンプトに対して `h` とタイプすると

```
The package fred has already been loaded with options:
 [foo]
There has now been an attempt to load it with options:
 [baz]
Adding the line:
 \usepackage[foo,baz]{fred}
to your document may fix this.
Try typing <return> to proceed.
```

と応答します。訂正するためには、そこで示されている通り、異なるオプションを読み込みます。L^AT_EX パッケージは、他のパッケージを呼び出すこともできるので、明示的に同じパッケージを 2 回呼び出さなくてもオプションの衝突が起こることがあります。

`Command <command> not provided in base NFSS.`

LaTeXe ではデフォルトでは `<command>` コマンドは与えられていません。このエラーは、次のコマンドの一つを呼び出すと発生します。

```
\mho \Join \Box \Diamond \leadsto
\sqssubset \sqsupset \lhd \unlhd \rhd \unrhd
```

これらは現在では `latexsym` パッケージの一部です。解決策は、ドキュメントのプリアンブルで、次のようにして加えることです。

`\usepackage{latexsym}`

LaTeX2e command `<command>` in LaTeX 2.09 document.

`<command>` は $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$ コマンドの一つですが、これは $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2.09$ ドキュメントです。解決策は、コマンドを $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2.09$ のコマンドに置き換えることか、あるいは、4.3 節で説明したようにネイティブモードでドキュメントを処理することです。

NFSS release 1 command `\newmathalphabet` found.

コマンド `\newmathalphabet` が新しいフォント選択スキームのリリース 1 で使われてきましたが、これは `\DeclareMathAlphabet` で置き換えられました。使い方は $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$ *Font Selection* を参照してください。

一番よい解決策は、`\newmathalphabet` コマンドを含んでいるパッケージを更新することです。新しいパッケージを探すか、(あなたが作者ならば) $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$ *Font Selection* を参照して新しいコマンドを知ることです。

このパッケージの更新さればバージョンが見つからない時は、`newfont` あるいは `oldfont` パッケージを使って解決できます。これは $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ に対して、`\newmathalphabet` のどのバージョンをエミュレートするかを伝えます。

つぎのような構文

`{\cal A}`, etc.

でドキュメントの数学フォントで選択しているのなら `oldfont` を使うべきです。ドキュメントの構文が

`\cal{A}`, etc.

であるならば、`newfont` を使います。

Text for `\verb` command ended by end of line.

`\verb` コマンドが始まっていますが、この行の終わっていません。これは `\verb` コマンドの末端文字を忘れていているのです。

Illegal use of `\verb` command.

`\verb` コマンドは別のコマンドの引数で使われています。これは $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ では許されていません。警告なく間違った出力を生成します。 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$ はエラーメッセージを出します。

6.2 古い内部コマンド

L^AT_EX 2.09 の多くの内部コマンドは削除されましたので、それらの機能は異なる方法で行われています。新しいことやクラスとパッケージ作成者のためのサポートされているインタフェースについては *L^AT_EX 2_ε for Class and Package Writers* に詳しく述べてあります。

```
\tenrm \elvrn \twlrm ...
\tenbf \elvbf \twlbf ...
\tensf \elvsf \twlsf ...
:
```

これらのコマンドは L^AT_EX 2.09 でプレロードされる 70 個のフォントへのアクセスを提供していましたが、対照的に L^AT_EX 2_ε では、14 個のフォントをプレロードして、メモリを節約します。この結果、上のコマンドを使ってフォントに直接アクセスする L^AT_EX ファイルは、もはや機能しません。

通常、次のようなエラーメッセージが 1 つ使われます。

```
! Undefined control sequence.
1.5 \tenrm
```

解決策は、L^AT_EX 2_ε で使われるフォント変更コマンドを使うようにドキュメント更新することです。詳しいことは *L^AT_EX 2_ε Font Selection* に説明されています。

これが不可能な場合は、最後の手段として、`rawfonts` パッケージを使うことができます。これは 70 個の L^AT_EX 2.09 フォントを読み込み、それらに古いコマンドで直接アクセスします。これには時間とメモリの両方が必要となります。70 個のフォントすべてをロードしたくない場合は、`rawfonts` を `only` オプションを使用すれば、必要なフォントを選択できます。たとえば、`tenrm` と `tenbf` が必要なら、

```
\usepackage[only,tenrm,tenbf]{rawfonts}
```

と書きます。

`rawfonts` パッケージはツールソフトウェアとして L^AT_EX の一部として配布されています。2.4 節を参照してください。

6.3 古いファイル

L^AT_EX を実行する上でよくある間違いの 1 つは、新しいバージョンではなく古いバージョンのパッケージを読み込んでしまうことです。標準パッケージから理解できないエラーメッセージが表示された場合は、パッケージの最新バージョンをロードしているかどうかを確認してください。ロードされたパッケージのバージョンは、ログファイルに次のように表示されるので、確認できます。

Package: fred 1994/06/01 v0.01 Fred's package.

適切な最新のドキュメントクラスあるいはパッケージを使ったかどうかは`\documentclass`と`\usepackage`に`<release-date>`オプションを使うことではっきりできます。この方法は、古いソフトウェアを使っているかもしれない他の人や組織にドキュメントを送る時に便利です。

6.4 さらに助けが必要な場合

遭遇した問題の解決策が見つからない場合は、*LaTeX: A Document Preparation System* あるいは *The LaTeX Companion* の中を探してください。LaTeX のインストールに問題があるのならば、配布ファイルに付属するインストールガイドをみてください。

それでもうまくいかない時、身近なところにいる LaTeX の専門家や LaTeX のメーリングリストにあたってください。

あなたがバグを発見したと思うなら、それを報告してください！まず最初に、問題が第三者のパッケージまたはクラスであるかどうかを調べます。2 節に記載されていないパッケージやクラスに引き起こされた問題であれば、LaTeX3 プロジェクトチーム宛ではなく、そのパッケージまたはクラスの作成者に報告してください。

バグが本当に LaTeX の中にあるのならば、問題を明確に示す短くて自己完結したドキュメントを作成してください。そのファイルを最新の（1 年以内の）バージョンの LaTeX で実行し、そして `latexbug.tex` を LaTeX で実行する必要があります。こうすると、`latexbug.tex` あるいは `bugs.txt` に記載されている LaTeX バグ用のアドレスに送るべきエラーレポートが、サンプルドキュメントとログファイルとともにできます。

7 楽しんで！

新しい企画の LaTeX 楽しんでくれることを望みますが、これが不可能な場合は、このドキュメントが助けになり、うまくいくことを望みます。

LaTeX が、あなたの役に立ち、プロジェクトチームの仕事をサポートしたくなったら、1.2 節を読んで実践的な方法を見つけてください。

References

- [1] Michel Goossens, Sebastian Rahtz and Frank Mittelbach. *The LaTeX Graphics Companion*. Addison-Wesley, Reading, Massachusetts, 1997.
- [2] Michel Goossens and Sebastian Rahtz. *The LaTeX Web Companion*. Addison-Wesley, Reading, Massachusetts, 1999.

- [3] Donald E. Knuth. *The T_EXbook*. Addison-Wesley, Reading, Massachusetts, 1986. Revised to cover T_EX3, 1991.
- [4] Leslie Lamport. *L^AT_EX: A Document Preparation System*. Addison-Wesley, Reading, Massachusetts, second edition, 1994.
- [5] Frank Mittelbach and Michel Goossens. *The L^AT_EX Companion second edition*. With Johannes Braams, David Carlisle, and Chris Rowley. Addison-Wesley, Reading, Massachusetts, 2004.