

pL^AT_EX ニュース 第 c14 号

2020 年 04 月 発行

作成者：日本語 T_EX 開発コミュニティ (<https://texjp.org/>)

コミュニティ版 pL^AT_EX 2_ε <2020-02-02> および <2020-04-12> について、pL^AT_EX 2_ε <2019-10-01> からの更新箇所をまとめます。

1 L^AT_EX 2_ε <2020-02-02> 対応

→参考：texjporg/platex#88

L^AT_EX 2_ε <2020-02-02> で NFSS (フォント選択の仕組み) が大幅に拡張されたことを受け、pL^AT_EX 2_ε <2020-02-02> で対応を施しました。シリーズ・シェイプの多軸化は fontaxes から、ファミリーごとの実シリーズ値は mweights から、それぞれアイデアを取り入れたようです。Itnews31 も参照してください。

pL^AT_EX 2_ε での拡張部分は以下のとおりです。

シリーズの多軸化

- `\kanjiseriesseries, \romanseriesseries` (`, \fontseries`)
`\DeclareFontSeriesChangeRule` によるシリーズ更新規則に従うようにした。
- `\kanjiseriesseriesforce, \romanseriesseriesforce`
無条件にシリーズを更新する (新規)。
`\fontseriesforce` は和欧文両方に影響。

シェイプの多軸化

- `\kanjishape, \romanshape` (`, \fontshape`)
`\DeclareFontShapeChangeRule` によるシェイプ更新規則に従うようにした。
- `\kanjishapeforce, \romanshapeforce`
無条件にシェイプを更新する (新規)。
`\fontshapeforce` は和欧文両方に影響。

総称ファミリーごとの実シリーズ値の設定

- `\DeclareFontSeriesDefault`
第一引数 (オプション) の総称ファミリー名に `rm, sf, tt` に加え `mc, gt` も指定できるようにした。

強調書体指定の一般化

- pL^AT_EX で再定義している `\em` (`, \emph`) でも `\DeclareEmphSequence` による入れ子の強調書体のカスタマイズ機能をサポートした。

既定値 L^AT_EX 2_ε に追従して pL^AT_EX 2_ε でも調整。

まず、L^AT_EX 2_ε での既定値の変更をみます。

- `\bfdefault: bx → b` に変更
- `\updefault: n → up` に変更
- `\shapedefault: \updefault → n` に変更

このうち「`bx → b`」は太字を原則 `b` (bold) に変更することを意味します。これだけでは互換性に関して懸念がありますが、L^AT_EX 2_ε では同時に「総称ファミリーごとの実シリーズ値の設定」の新機能を活用して

- Computer Modern と Latin Modern の場合だけは従来どおり太字を `bx` (bold-extended) に維持

という挙動を実現しています。

pL^AT_EX 2_ε では、`\bfdefault` と `\updefault` について L^AT_EX 2_ε の設定がそのまま和文にも引き継がれます。残る一つは和文用の既定値が別に存在します。

- `\kanjishapedefault: \updefault → n` に変更

さて、和文の太字は従来 `\bfdefault` 一択 (結果的に `bx`) でしたが、pL^AT_EX 2_ε <2020-02-02> では「総称ファミリーごとの実シリーズ値の設定」の新機能が和文ファミリー `mc, gt` にも使えます。ここで、`mc, gt` の太字をどう設定するかは議論の余地があります：

1. 互換性のため、従来どおり `bx` を維持する。
2. Computer Modern でも Latin Modern でもないので、`b` に変更する。

この 2 案のうち、現在の版では 2. すなわち

- 和文ファミリー `mc` と `gt` の太字も `b` (bold) に変更を採っています。これは、和文の太字は線が太いだけで、横幅が広がるわけではないためです。

2 NFSS の和文対応の改善

pL^AT_εX 2_ε <2020-04-12> での修正です。L^AT_εX 2_ε <2020-02-02> での NFSS の拡張とは無関係に、従来から pL^AT_εX 2_ε の NFSS に存在した和文フォントでの不具合に対処しました。

`\fontshape` で和文シェイプが未定義なら維持 端的には `{\itshape a}` のようなシェイプ変更で

```
Font shape 'JT1/mc/m/it' undefined
using 'JT1/mc/m/n' instead on ....
Font shape 'JY1/mc/m/it' undefined
using 'JY1/mc/m/n' instead on ....
```

という LaTeX Font Warning が出るのを抑制することを目的とした修正です。→ [texjporg/latex#90](https://texjporg.org/latex#90)

pL^AT_εX 2_ε の `\fontshape` は、欧文に加えて和文のシェイプも変更するように再定義されていました (アスキー pL^AT_εX 2_ε の仕様)。しかし、これでは `\itshape` のような命令で余分な警告を発してしまいます。

- L^AT_εX 2_ε が定義する `\itshape` などのシェイプ変更命令は、内部で `\fontshape` を呼び出す。
- pL^AT_εX 2_ε の `\fontshape` は、欧文書体だけでなく和文書体のシェイプも変更しようとする。
- しかし、和文書体のシェイプはほとんど “n” しか用いられず、`\DeclareFontShape` での定義も “n” しか与えられないことが多い。
- 結果的に、欧文書体のシェイプを変更するつもりでも「和文書体のシェイプが未定義」という警告が出てしまう。

これを改善するため、pL^AT_εX 2_ε <2020-04-12> では

- `\fontshape` が選択しようとした和文のシェイプが未定義の場合は、和文のシェイプは変更せず、欧文のシェイプのみを変更する

という仕様に変更しました。これは `\fontshapeforce` も同様です。

なお、`\kanjishape` や `\kanjishapeforce` は和文書体のシェイプを変更する意図でしか実行されないため、未定義かどうかは確認しません。また、`\usefont` は和欧文両用ですが、一緒に指定されたエンコーディングに応じて `\useroman` と `\usekanji` の一方だけが実行されることが明白なため、やはり未定義かどうかは確認しません。

`sub`, `ssub` の和文対応 `\DeclareFontShape` で使われる `size function` の `sub`, `ssub` が和文フォントを正しく取得できない不具合を修正。

これが露呈する具体的な症例としては、GitHub の [texjporg/jsclasses#53](https://github.com/texjporg/jsclasses#53) で報告されている「日本語の数式ファミリーを宣言した状態で `bm` パッケージを使うとゴシックにならない」がありましたが、今回のリリースで解消しました。

`\normalfont` 末尾の `\ignorespaces` 削除 オリジナルの L^AT_εX 2_ε の 1995 年の修正に追随し、pL^AT_εX 2_ε の `\normalfont` 末尾になぜか残っていた `\ignorespaces` を削除しました。

`\KanjiEncodingPair` 不在のエラー改善 2004 年のアスキー pL^AT_εX 2_ε で、`\KanjiEncodingPair` による和文縦横エンコードのセット化が必須となっています。しかし、セット化せずにエンコードを定義した場合のエラーが判りづらかったため、改善しました。(#92)

`\DeclareKanjiSubstitution` の修正 和文の代用書体の宣言が欧文の `\DeclareFontSubstitution` と干渉し、誤ったフォント定義ファイルが読まれようとする問題を解消しました。(#93)

`\verb` の改良 `\verb+>+` を和文で挟むと後ろだけアキが入って前には入らない現象に対処し、`\texttt{>}` と同様に前後にアキが入るようにしました。(#87)

3 開発版のテストのお願い

今後 pL^AT_εX に導入するかもしれない修正パッチや仕様変更のテストにご協力ください。T_EX ファイルの冒頭 (`\documentclass` より前) で

```
\RequirePackage{exppl2e}
```

と書くことで、現在の開発版をテストすることができます。詳細は [exppl2e.pdf](https://github.com/texjporg/exppl2e) を参照してください。ここには、その他の pL^AT_εX 2_ε の既知の制約事項も記載しています。T_EX Forum や GitHub の Issue でのバグ報告やご意見を歓迎します。

- <https://github.com/texjporg/platex>
- <https://github.com/texjporg/uplatex>