

# Package `plautopatch` v0.2

Hironobu Yamashita

2018/08/21

日本の `pLATEX`/`upLATEX` フォーマットや専用パッケージが、これらを知らない `LATEX` パッケージ（しばしば海外で作られた汎用のもの）と衝突することがあります。最悪の場合にはエラーが出たり、誤った出力が得られたりすることがあります。

この `plautopatch` の目的は、こうした非互換を意識せずに済むようにすることです。具体的には、`pLATEX`/`upLATEX` と衝突するパッケージが使われた場合に、その衝突を解消するパッチを提供するパッケージを必要に応じて自動的に読み込みます。こうすることで、ソースコードを簡潔にできるだけでなく、`pLATEX`/`upLATEX` で動作するソースと通常の `LATEX` ソースの見た目を近づけることができます。

このパッケージは GitHub で開発しています。

<https://github.com/aminophen/plautopatch>

## 動作条件

このパッケージは `filehook` パッケージ（Martin Scharrer 氏の作）に依存します。

## 使い方

このパッケージを `LATEX` ソースの冒頭で読み込みます。このために、`\RequirePackage{plautopatch}` を `\documentclass` や他のコマンドよりも前に読み込むことをお勧めします（クラスファイルなどが問題のあるパッケージを読み込む可能性もあるため）。

例を示します。

```
\RequirePackage{plautopatch}
\documentclass{tarticle}% 縦組クラス (plext 使用)
\usepackage{array}% plext と非互換
\begin{document}
...
\end{document}
```

上記の例では、`tarticle` クラスが内部で読み込む `plext` パッケージと、ソース中で `\usepackage` している `array` パッケージが衝突してエラーになる場合があります。しかし、冒頭で `\RequirePackage{plautopatch}` とだけ書いておけば、`array` パッケージの時点で `plextarray` パッケージが追加で読み込まれるため、問題が解消します。このように自動追加されたパッケージは、`\end{document}` の時点で次のように一覧として表示されま

す（複数の場合はコンマと空白で区切ったリストになります）。

```
**** List of packages loaded by 'plautopatch': ****  
plextarray.  
*****
```

## 現在対応しているパッケージの一覧

凡例：

- <元のパッケージ> (<元が含まれるバンドル名>)  
 <パッチのパッケージ> (<パッチが含まれるバンドル名>)

現在のバージョン (2018/08/21 v0.2) がサポートしているのは下記のパッケージです。

- tracefnt (latex)  
 → ptrace/uptrace (platex/uplatex)
- fltrace (latex)  
 → pfltrace (platex)
- array (latex-tools)  
 → plarray (platex-tools)
- array (latex-tools) + plect (platex)  
 → plectarray (platex-tools)
- delarray (latex-tools) + plect (platex)  
 → plectdelarray (platex-tools)
- everysel (ms)  
 → pxeverysel (platex-tools)
- everyshi (ms)  
 → pxeveryshi (platex-tools)
- atbegshi (oberdiek)  
 → pxatbegshi (platex-tools)
- ftnright (latex-tools)  
 → pxftnright (platex-tools)
- pdfpages  
 → pxpdfpages (maintained here!)

もちろん、このリストは随時、追加・削除・置き換えていく予定です。互換性の問題や追加したいパッケージがある場合はご一報ください。

## 特定のパッケージを除外したい場合

デフォルトでは、上記のリストに登録されている<元のパッケージ>が使われたことを検出すると、全て自動的にパッチを読み込みます。しかし、時にはこれが逆効果となり、問題が起きる可能性は否定できません。そのような場合は

```
\plautopatchdisable{<元のパッケージ>}
```

と書くことで、そのパッケージを検出対象から除外します。複数ある場合は

```
\plautopatchdisable{<元のパッケージ 1>,<元のパッケージ 2>}
```

のようにコンマで区切っていくつでも除外できます。