

\footnote再考

鹿野 桂一郎

ラムダノート株式会社

k16.shikano@lambdanote.com
@golden_lucky

2023 年 11 月 11 日
於 TeXConf 2023

\footnoteの挙動が気に食わない

- ▶ ボックスの中で使ってもページ下部に出てほしい
- ▶ ボックスが改ページしてもページ下部に出てほしい
- ▶ 脚注にも脚注を付けたい

自分で作ろう！ でもどうやって？

- ▶ $\text{T}_{\text{E}}\text{X}$ の脚注は「インサート」という謎機能
- ▶ 直感的に思いつく素直な実装方法は、むしろこんな感じではないだろうか？



1. 脚注の内容のみをブロック要素として組版
2. アンカーのあるインライン要素に紐づける形で、そのブロック要素をいったん保持
3. ページ分割時に、もしアンカーがあったら、紐づけられているブロック要素の高さを考慮する
4. そのページの下部に、そのブロック要素を配置する

T_EX だけでは（たぶん）無理

- ▶ 少なくとも、インサートを使う方法では無理そう（後述）
- ▶ 行分割やページ分割に割り込んでレジスタやトークンリストの操作ができるようになれば、あるいは…
- ▶ そういえば最近の L^AT_EX カーネルには「フック」とかいう機能が導入されていたような

LaTeXのフックでも（たぶん）無理

- ▶ TeX エンジンレベルの処理には割り込めない
- ▶ LaTeX ラボ[1]には`\footnote`に関するフックもあるが、これらの対象は事実上`\@footnotemark`と`\@footnotetext`なので、結局はインサートとして読まれてしまってダメ（後述）

[1] <https://ctan.org/tex-archive/macros/latex/required/latex-lab>

そこで LuaTeX

- ▶ Lua のオブジェクトとして構文木（っぽいもの）が作られる
 - ▶ node list と呼ばれている[2]
 - ▶ node list に対する操作として、行分割処理の前後などで発動するフックが書ける[3]
- ▶ `\attribute` という新しいプリミティブが使える
 - ▶ Lua のオブジェクトと TeX のコードとの間で、メタ情報をやり取りできる[4]

[2] <https://ctan.org/pkg/nodetree>

[3] <https://wiki.luatex.org/index.php/Callbacks>

[4] <https://wiki.luatex.org/index.php/Attributes>

実装の方針

▶ `\footnote`を見たら以下をする。

1. `\footnote`の中身を取りあえず組む
2. その行の直下に配置
3. 脚注マークと紐づける
4. 組んだ要素の高さと深さをつぶす
5. その要素をページ下部へ

実装の方針

▶ `\footnote` を見たら以下をする。

1. `\footnote` の中身を取りあえず組む
2. その行の直下に配置 ← `post_linebreak_filter`
3. 脚注マークと紐づける ← `\attribute`
4. 組んだ要素の高さと深さをつぶす ← `vpack_filter`
5. その要素をページ下部へ ← `pre_output_filter`

post_linebreak_filter

```
push_footnotes_below_lines = function (head, group)
  for item in node.traverse_id(node.id("whatsit"), head) do
    local is_footnote = node.has_attribute(item, 100)
    if is_footnote and is_footnote > 0 then
      local footnote = node.copy(tex.box[is_footnote])
      head, new = node.insert_after(head, item, footnote)
      item = item.next
    end
  end
  return head
end

luatexbase.add_to_callback
("post_linebreak_filter", push_footnotes_below_lines, "pushftn")
```

Whatsitに\attribute[5]を付与

```
\def\yafootnote#1{\nobreak%
  \global\advance\yafootnotecount 1
  \global\expandafter\newbox
    \csname yafoot_\the\yafootnotecount\endcsname
  \begingroup
    \attribute100=\expandafter\the\csname
      yafoot_\the\yafootnotecount\endcsname
    \expandafter\yafootnotemark\expandafter{\the\yafootnotecount}
    \vadjust {\pdfliteral{}}%
  \endgroup
  \global\expandafter\setbox
    \csname yafoot_\the\yafootnotecount\endcsname
    \vtop{\yafootnotetext{#1}}}
```

[5] この手法についてはTUGboat, Volume 31 (2010), No. 3 の Paul Isambert による記事 <https://tug.org/tugboat/tb31-3/tb99isambert.pdf> が参考になる。

post_linebreak_filter 後

横書き書籍においては、本文の任意の箇所に対して補足的な情報を付記したい場合、その本文と同一ページの最下部に「脚注」^{†1}として補足情報を組版することが多い。

^{†1} ここでは、補足情報をページの左右余白に配置する「傍注」、章末や巻末に配置する「後注」、行間に配置する「行間注」^{†2}などと「脚注」を区別していることに注意。

^{†2} 段落末に配置する「段落注」も行間注の一種とみなせる。

T_EX 系の組版システム ^{†3} では脚注を簡単に利用できる。とくに L^AT_EX で用意されて ^{†3} 以降では単に「T_EX」と表記する。

いる標準的な `\footnote` コマンドは、さまざまなパラメーターを設定するだけで、脚注に対する組版上のさまざまな要件を簡単にカスタマイズできるようになっている。

vpack_filter

```
crush_height_of_vlist = function (head, group, size)
  for list in node.traverse_id(node.id("hlist"), head) do
    for item in node.traverse(list) do
      local f = node.has_attribute(item, 200)
      if f then
        item.height = 0
        item.depth = 0
      end
    end
  end
  return head
end

luatexbase.add_to_callback
("vpack_filter", crush_height_of_vlist, "crushvbox")
```

vpack_filter 後

横書き書籍においては、本文の任意の箇所に対して補足的な情報を付記したい場合、その本文と同一ページの最下部に「脚注」^{†1}として補足情報を組版することが多い。

^{†1} ここでは、補足情報をページの左右余白に配置する「傍注」、章末や巻末に配置する「後注」、行間に配置する「行間注」などと「脚注」を区別していることに注意。
^{†2} TeX 系の組版システム^{†3}では脚注を簡単に利用できる。とくに L^AT_EX で用意されている標準的な `\footnote` コマンドは、さまざまなパラメーターを設定するだけで、脚注に対する組版上のさまざまな要件を簡単にカスタマイズできるようになっている。

pre_output_filter

```
move_footnote_bottom = function (page_head, group, s)
  local yaftnins = node.new("vlist")
  local n_head = node.copy_list(page_head)
  recur = function (n)
    for list in node.traverse(n) do
      local footnotebox = node.has_attribute(list, 200)
      if footnotebox then
        footnote = node.copy(tex.box[footnotebox])
        for ftnitem in node.traverse(footnote.head) do
          if node.has_attribute(ftnitem, 200) then
            footnote.head = node.remove(footnote.head, ftnitem)
          end
        end
        if yaftnins then
          yaftnins.list, new = node.insert_after(yaftnins.list, yaftnins.tail, footnote)
        end
        n_head = node.remove(n_head, list)
        n_head = recur(list.head)
      elseif list.head then
        n_head = recur(list.head)
      end
    end
    return n_head
  end
  page_head = recur(n_head)
  if yaftnins.list then
    tex.box.footins = node.copy(node.vpack(yaftnins.list))
  end
  return page_head
end
```

pre_output_filter 後

横書き書籍においては、本文の任意の箇所に対して補足的な情報を付記したい場合、その本文と同一ページの最下部に「脚注」^{†1}として補足情報を組版することが多い。

$\text{T}_{\text{E}}\text{X}$ 系の組版システム^{†3}では脚注を簡単に利用できる。とくに $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ で用意されている標準的な `\footnote` コマンドは、さまざまなパラメーターを設定するだけで、脚注に対する組版上のさまざまな要件を簡単にカスタマイズできるようになっている。

^{†1} ここでは、補足情報をページの左右余白に配置する「傍注」、章末や巻末に配置する「後注」、行間に配置する「行間注」^{†2}などと「脚注」を区別していることに注意。

^{†2} 段落末に配置する「段落注」も行間注の一種とみなせる。

^{†3} 以降では単に「 $\text{T}_{\text{E}}\text{X}$ 」と表記する。

なぜ通常の\footnoteではこれが…

- ▶ \footnoteはインサートで実装されている
- ▶ どのモードであれ、インサートは**周囲**の垂直リストに入る
- ▶ つまり、もし内部垂直モードなら、**メイン**垂直リストには入らない
- ▶ ページ作成機能が\box255に移動するのは、**メイン**垂直リストの要素だけ
- ▶ つまり、内部垂直モードの内側にインサートを置いても、ページ作成機能には届かない！[6]

[6] “The TeXbook” には明記されていないようだが “TeX by Topic” などでは説明されている。

原理的にボックス内でも問題ない

横書き書籍においては、本文の任意の箇所に対して補足的な情報を付記したい場合、その本文と同一ページの最下部に「脚注」^{†1}として補足情報を組版することが多い。

TeX系の組版システム^{†3}では脚注を簡単に利用できる。とくにL^AT_EXで用意されている標準的な`\footnote`コマンドは、さまざまなパラメーターを設定するだけで、脚注に対する組版上のさまざまな要件を簡単にカスタマイズできるようになっている。

^{†1} ここでは、補足情報をページの左右余白に配置する「傍注」、章末や巻末に配置する「後注」、行間に配置する「行間注」^{†2}などと「脚注」を区別していることに注意。

^{†2} 段落末に配置する「段落注」も行間注の一種とみなせる。

^{†3} 以降では単に「TeX」と表記する。

でもページ分割があると…

横書き書籍においては、本文の任意の箇所に対して補足的な情報を付記したい場合、その本文と同一ページの最下部に「脚注」†1として補足情報を組版することが多い。

TeX系の組版システム†3では脚注を簡単

†1 ここでは、補足情報をページの左右余白に配置する「傍注」、章末や巻末に配置する「後注」、行間に配置する「行間注」†2、カレレ「脚注」

に利用できる。とくに \LaTeX で用意されている標準的な`\footnote`コマンドは、さまざまなパラメーターを設定するだけで、脚注に対する組版上のさまざまな要件を簡単にカスタマイズできるようになっている。

(tcolorboxの場合の) 原因と対処

- ▶ tcolorbox は $\text{T}_{\text{E}}\text{X}$ のページ作成機能ではなく、`\vsplit` を使ってページ分割を手書きしているの
で[7]、`footins`の高さが考慮されない
- ▶ Lua 側で「つぶした高さ」を保存しておき、 $\text{T}_{\text{E}}\text{X}$ 側で
用意したディメンジョン (`\my@tcb@ftn@height`)
経由で渡して、`\vsplit` の計算時に考慮するように
改造すれば…！

[7] <https://github.com/T-F-S/tcolorbox/blob/18aecbbacc61445c178c3e9a35cad588b3665bf/tex/latex/tcolorbox/tcbbreakable.code.tex#L394>

tcolorbox の分割計算を改造

```
\newdimen\my@tcb@ftn@height
\my@tcb@ftn@height\z@

\def\tcb@vsplit@upper{%
  \tcbdimto\tcb@split@dim{\tcb@split@dim-\my@tcb@ftn@height}
  \global\my@tcb@ftn@height\z@
  \setbox\tcb@upperbox=\vsplit\tcb@totalupperbox to\tcb@split@dim%
  \edef\tcb@upper@box@badness{\the\badness}%
}
```

- ▶ Lua 側ではフック `buildpage_filter` を使うことで `\my@tcb@ftn@height` を更新
- ▶ 必要な高さを事前計算するため、`lualatex` を複数回実行

顧客が本当に欲しかったもの

横書き書籍においては、本文の任意の箇所に対して補足的な情報を付記したい場合、その本文と同一ページの最下部に「脚注」^{†1}と

^{†1} ここでは、補足情報をページの左右余白に配置する「傍注」、章末や巻末に配置する「後注」、行間に配置する「行間注」^{†2} などと「脚注」を区別していることに注意。

^{†2} 段落末に配置する「段落注」も行間注の一種とみなせる。

して補足情報を組版することが多い。

TeX 系の組版システム^{†3}では脚注を簡単に利用できる。とくに \LaTeX で用意されている標準的な`\footnote`コマンドは、さまざまなパラメーターを設定するだけで、脚注に対

^{†3} 以降では単に「TeX」と表記する。

する組版上のさまざまな要件を簡単にカスタマイズできるようになっている。

課題

- ▶ ページ分割ボックスの実装方法に応じた個別対応が必要
- ▶ アウトプットルーチンを書き換える必要がある**場合がある**
- ▶ 浮動要素が同じページに出現する場合の挙動がむずい（最悪、同じ脚注が2回出る）
- ▶ Beamer 非対応（ボックス内で独自に脚注を出力させているため）
- ▶ 空いている`\attribute`を \TeX に生成させたいが、それを Lua 側から知る方法がわからない

でもすぐに使ってみたい！

- ▶ 第三者につるしで使ってもらうには、もう一息いろいろ整備が必要
- ▶ ラムダノート発行の書籍ではすでに使っています[8]
- ▶ ソースはGitHubにあるので使ってみることは可能です
 - ▶ <https://github.com/k16shikano/yafootnote>

[8] 株式会社 CARTA HOLDINGS 監修、和田卓人 編『事業をエンジニアリングする技術者たち』など。<https://www.lambdanote.com/collections/carta>