

Chapter 17 Paragraph End

17.1 パラグラフはどこで終わるか

- `\par` を直書きした場合
- 空行
暗黙のうちに `\par` トークンが挿入される
- 非制限水平モード中の垂直コマンド
暗黙のうちに `\par` トークンが挿入された上に、当該垂直コマンドが読み直される。例を以下に示す。

```
Former_paragraph, former_paragraph, last_words.\vskip6pt  
A_new_paragraph_is_starting_here.
```

(出力)

```
Former paragraph, former paragraph, last words.
```

```
A new paragraph is starting here.
```

★ `\par` によらない Paragraph End もある！

- `\vbox{……}` の閉じかっこ
- `\insert{……}` の閉じかっこ
- `\output={……}` の閉じかっこ

これらは、

<https://tex.stackexchange.com/questions/37042/does-tex-always-insert-the-macro-par>
の通り、`\par` によらない。T_EXweb を読めばわかる (らしい) が、module §1096 の `end_graf` を直接実行している模様。

17.1.1 `\par` トークンと `\par` プリミティブの用語の使い分け

以下の例文を読めばわかるだろう。

- `\par` トークンは自動で挿入される。
- `\par` プリミティブが実行されると段落終了処理が行われる。

17.1.2 `\parfillskip`

- `\parfillskip` は段落の最後に入る水平グルー (plain T_EX デフォルトでは 0 pt plus 1 fil)
- 段落の最後まで来ると、`\par` より前に、`\unskip\penalty10000\hskip\parfillskip` に相当するものが挿入される。`\penalty10000` の代わりに `\nobreak` であるという解釈も可
- 以下、挿入の例を一つ示す

```
Former_paragraph, end.
```

(空行)

```
\noindent_Begin_new_paragraph.
```

↓ (空行 + `\noindent` が `\par` に変わる)

Former_paragraph, _end. _\par _Begin _new _paragraph.

↓ (\par 前に \unskip\penalty10000\hskip\parfillskip を挿入する)

Former_paragraph, _end. _\unskip\penalty10000\hskip\parfillskip\par _Begin _new _paragraph.

↓ (_\unskip が消滅)

Former_paragraph, _end. \penalty10000\hskip\parfillskip\par _Begin _new _paragraph.

- \parfillskip は plain T_EX デフォルトで 0pt plus 1fil であるので、次の2つのコードの結果に次のような差が出る。

\hfil の場合 : \hfil vs \hfil で釣り合い、bullet が中点へ。

\hfill の場合 : \hfill vs \hfil で \hfill の方が強いので、bullet が右端へ。

Old_line, _former_line, _here. _\hfil\$\bullet\$\par _New_line.

(出力)

Old line, former line, here. •
New line.

Old_line, _former_line, _here. _\hfill\$\bullet\$\par _New_line.

(出力)

Old line, former line, here. •
New line.

17.2 Assorted remarks

17.2.1 グループ末で改行する時には場所を選んで \par の作用点に注意する話

- \baselineskip や \leftskip など、改段落に影響を与えるパラメタをグループ内でいじっている時、それを改段落に反映させるためには、グループ内で \par する必要がある。

\leftskip2zw{\leftskip1zw_Old_line, _here. \par} _New_line. \par

(出力)

Old line, here.
New line.

\leftskip2zw{\leftskip1zw_Old_line, _here.} \par _New_line. \par

(出力)

Old line, here.
New line.

- グループ内で \medskip など vertical command を使い、\par を挿入させる hack も可能。

\leftskip2zw{\leftskip1zw_Old_line, _here. \medskip} _New_line. \par

(出力)

Old line, here.
New line.

17.2.2 `\hfill\break\par` によるパラグラフ終了

Underfull 警告が出ます。plain T_EX でも L^AT_EX でも確かに出ました。`\hfill\break\hbox{ }\par` とかすれば OK。

17.2.3 9章にある Ending a paragraph with leaders について

- `\nobreak\leaders\hrule\hfill\par` では上手くいかない。
`\par` によって挿入された `\unskip` によって、leaders が表示されなくなる。
- `\nobreak\leaders\hrule\hfill\hbox{ }\par` と、保護するといける。

```
Old_line, _here.\nobreak\leaders\hrule\hfill\par_New_line.
```

(出力)

```
Old line, here.  
New line.
```

```
Old_line, _here.\nobreak\leaders\hrule\hfill\hbox{ }\par_New_line.
```

(出力)

```
Old line, here._____
New line.
```

17.2.4 `\nobreak` をうまい位置で使って、改段落時の改ページを抑制できる理由を説明

```
Old_line, _here.\par\nobreak\medskip_New_line.
```

(出力)

```
Old line, here.  
New line.
```

この時の vertical list は次のようになっている。

```
\hbox(6.94444+0.0)x ... % last line of paragraph
\penalty 10000 % \nobreak
\glue 6.0 plus 2.0 minus 2.0 % \medskip = \penalty10000のせいでここでの改行は無理
\glue(\parskip) 0.0 plus 1.0 % \parskip = 前が non-discardable item でない glue なので改行無理
\glue(\baselineskip) 5.05556 % interline glue = 前が non-discardable
item でない glue なので改行無理
\hbox(6.94444+0.0)x ... % first line of paragraph
```

17.2.5 `\parfillskip` の値を有限にすると

- 段落末の他のグルーが natural width から伸びる。
- それに引き続く display 数式において、`\abovedisplayshortskip` が選ばれず、必ず `\abovedisplayskip` が選ばれる。

```
\abovedisplayskip=0.5cm\abovedisplayshortskip=0cm
```

```
あいうえお。
\[
Y=3X
\]
```

```
\parfillskip=10cm
```

```
あいうえお。  
\[  
Y=3X  
\]
```

(出力)

あいうえお。

$$Y = 3X$$

あ い う え お 。

$$Y = 3X$$

17.2.6 段落頭が `\noindent` の時に注意すべきこと

- 前の段落末がちょうど右端まで行ってしまうと、段落が変わったことに気付けなくなってしまう
→ `\parfillskip=0cm plus 1fil` (デフォルト) を `\parfillskip=1cm plus 1fil` と変更すれば解決。必ず右端に 1cm 以上の虫食いができるので、改段落がわかる。

Chapter 18 Paragraph Shape

18.1 1 段落を構成する水平リスト

- `indentation box`
この幅は `\parindent` により指定される。plain TeX デフォルトは 20 pt。
- (中身)
- `\parfillskip` グルー

この水平リストが、`\hsize-\leftskip-\rightskip` の幅になるよう分割される。(詳細は行分割アルゴリズムにて)

- `\rightskip` に伸縮をもたせておくと、`\raggedright` ができる。
- Shape Parameters の `\parshape` や `\hangindent` は、`\hsize` 自体をいじることにより機能する。

18.2.1 ぶら下げ組と `\hangindent`

- `\hangindent`
デフォルトは 0pt = 右も左もぶら下げインデントを指定しない
正なら左ぶら下げインデント指定
負なら右ぶら下げインデント指定
- `\hangafter`
デフォルトは 1 = ぶら下げインデントは最初の 1 行だけ除外
正ならぶら下げインデント開始までに除外する行数指定
負ならぶら下げインデント終了までの行数指定

- ぶら下げインデント (`\hangindent`) は行頭インデント (`\parindent`) とは独立なので, `\hangafter` が 0 以下なら, 1 行目は両方が加算されうる。

→行頭インデントが不要な場合には `\noindent` で除去 (行頭インデントだけを除去できます)

★ `\hangindent` と `\hangafter` の値は, `\par` ごとにデフォルトに戻ります

→ `\everypar` を使わないと, 統一的にはいじれない

18.2.2 `\parshape`

- 最初の n 行について, インデント量と行長を指定する。
- $n + 1$ 行目以降が発生したら, n 行目の設定を受け継ぐ。
- `\par` ごとにリセット
- 一例を以下に示す

```
\parshape=7_6zw_28zw_5zw_29zw_4zw_30zw_3zw_31zw_2zw_32zw_1zw_33zw_0zw_34zw
\lipsum[1]
```

(出力)

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis.

Curabitur dictum gravida mauris. Nam arcu libero, nonummy

eget, consectetur id, vulputate a, magna. Donec vehicula augue

eu neque. Pellentesque habitant morbi tristique senectus et netus et

malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus

rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus

eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in,

pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum.

Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac,

nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis

nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci

sit amet orci dignissim rutrum.

18.3 Assorted remarks

ここから先は次回で