

2 New! 2015/07/02 (Thu)

to: grpE members and Others

kymst posted

山下です。もう少し、chibitex について、というか、emacs における major mode について話そうと思います。所詮は tips (豆知識, trivia) でしかない、とも言えますが、考え方によっては端末と付き合うための「実践理性」に対する批判にもなっています。この辺も emacs-lisp に絡めて連載ものにしていく積もりですので、お付き合い下さい。何しろ、病院、ヒマなんです.....

2.1 PLE relation

最終的な結論の先取りになりますが、端末に触れる以上、editor ^{*1}には詳しくなった方がいいです。何と言っても、pslptr ^{*2} の主目的が、(program の source file を代表とする) file を作成することである以上、その主役は editor 以外にないからです。その意味で、programmer と彼が操る program 言語との関係は対象言語 とその言語使用者 (および彼 / 彼女が考えている限りでの意味) という object-level にありますが、programmer \Leftrightarrow program 言語を編集する際に彼が用いる editor、略して「prgmr / editor 関係」は、meta-level の関係になります。multilingual-programmer が自分 / 言語 / editor という 3 項の関係を、扱う言語ごとに変えてはやってられません。第 3 項 editor 上で、最初の 2 項からなる複数個の pair が有関係的構造様式として体系化されるべきであると、ないし体系化されているのだ、と思います。

2.2 Emacs / Emacs-Lisp

emacs については、上記の editor 理解の上に立った上で、世界最強の editor である、と納得してくればそれで十分です。もともとは、MIT の物理学科にいた Richard Stallman という天才 programmer が、MIT の AI Labo にある computer TECO 上で動かすことから歴史に登場することになった editor で、名前の由来は Editing-MACroS だそうです。

他の editor との違いは、elisp という 1 つの interpreter programming 言語の 処理/実行系の full set を、マルのまま含んでいる、ということであり、そして更に、その処理/実行系が emacs で書かれていること、従って emacs で document を書くということは、elisp という programming 言語で document を書いていることに他ならない、ということ、です。解り難いですよ、もう少し整理します。

ここで以下、lower case で (番号があってもなくても) emacs と書かれたときには、大体におい

^{*1} この意味はいいですよ、ある程度の長さの document を編集するための soft ware のことを、今日では単に editor と呼びます。かつては、1 行ごとの編集が中心だったので、このような single-line-editor と区別して multiline-editor と呼ばれたり、あるいは kbd からの入力と同時に display 上に編集結果が反映されるので real-time editor とも呼ばれました。

^{*2} Personal Computer を以下こう表わします。パソコンという呼び方、キライです。pasokon \Leftrightarrow perso-com って、音の関係が何もない。

て editor としての Emacs と program 言語 elisp のの処理実行系を合わせたものを意味する、と
 考えて、それ程の支障はないと思われる。

- (1) document を書くことだけに、elisp という programming 言語を占有できる、ということ。
 つまりは、editor Emacs のオマケとしての elisp 実行環境。この Emacs を emacs0 と呼ぶ。
- (2) Emacs で document を書くに際して、editor としての振舞いが気に入らないので、elisp で
 書いた program と取り換えて、それが走るようにしちゃおう。この Emacs を emacs1 と
 呼ぶ。

- (3) では、emacs0 を emacs1 にする program は、どこで、どのように書かれるか?

答え: emacs0 上で書かれる。Editor emacs0 は (emacs1 もだが), elisp で programming を
 行なうための front-end, つまりは program を書きそれを実行するための下請け application
 でしかない。

この意味は、editor Emacs は elisp での programming のための user interface であり、更
 に言えば、editor Emacs は program 実行処理系 elisp のオマケである。

- (4) さて、emacs0 上で書かれた program によって、emacs0 は目出たく emacs1 に換骨奪胎し
 た、としよう。特に YaTeX を使っている member には解りやすいと思うが、YaTeX mode
 というのがこの emacs1 に当たる。
- (5) 人間、嗚呼、この欲深きものよ。いつかは、YaTeX にも気に入らない挙動が目につくよう
 になる。もちろん elisp で program を書いてそれを Emacs で実行しよう。この Emacs は
 emacs2 であることに異論はないであろう。

ただ、もう 1 つ重要な点がある。YaTeX そのものが elisp で書かれていることは極めて重
 要である。

- (6) 世に、「カスタマイズ可能」という触れ込みの application は星の数ほどあるが、カスタ
 マイズするための macro 記述言語が、その application の program source が書かれた言語
 であることはほぼない。最近、python でこうした movement があることを知ったところ
 である。つまり、macro 記述に expertize できたとしても、その application 自体について勉
 強できたことにはならないのである。例えば「秀丸」editor は C/C++ で書かれている (ら
 しい) が、macro は C/C++ で書かれるのではない。

Emacs は違う。elisp program を書く筆記用具としての、editor としての Emacs は elisp
 で書かれて (= program されて) いる。その上で走る program の 1 つの代表としての
 YaTeX ももちろんそうである。これまた星の数ほどある、Emacs の major mode は、すべ
 て elisp によって書かれている。

- (7) ということは、ある major mode μ -mode が emacsN によって Hyper- μ_N -mode から
 HyperHyper- μ_{N+1} -mode に書き換えられるとき、処理系、実行系としての emacs は
 emacs0 のままであることになる。

高次の実行系としての emacsN は、それが内包する高次の elisp program である major
 mode μ_N で書かれ、 μ_{N+1} を記述しかつそれを動かすが、 μ_{N+1} は最下層にある elisp に

よって「動かされている」のである。

- (8) この構造は、現代の von Neumann 型計算機の言語階層、つまり Machine Language < Assemble Lang < Compile Lang における、下方向非透明性 ないしは 意識的忘却、に似ている。Compile 言語で program を書くとき、それが機械語ではどうなるか? を考える必要はない (多少の例外、極端な計算速度が要求されるような場合、など、もあるが)。

その意味で、「Emacs は editor じゃない。あれは OS だ」とか、「Emacs を使いこなすのに Emacs を好きになる必要はない。好きな Emacs にすればいいだけさ」とか、言われるのも頷ける。要は、いくら高次の level で動くように major-mode を書いたところで、いざとなれば最下層 elisp を調べればいいだけの話なのだ。この雑草のような flexibility が、また Emacs の魅力であるように思える。

2.3 picoTeX-mode

...にまで話が辿り着きませんでした。連載なので、これからも楽しみにしてくれると嬉しいです。また、反論、疑問、質問、イチャモンなど頂けると、もっと嬉しいです。

今回は、picoTeX における enumeration environment についての話から始めましょう。箇条書きですね。

TeX では

```
\begin{enumerate}
\item hoge
\item hero
\end{enumerate}
```

みたいなヤツです。

PS.

この file 20150702log.tex 自体が、chibitex-mode で書かれた file 20150702log.chx から、やはり chibitex-mode の 1 つの module である chx2TeX translator という elisp program で (半分) 自動生成されたものです。もし興味のある方がいらっしゃいましたら、chx と tex file をいっしょにして送ります。

ただ、この辺については chibitex の設計思想が脆弱だった (なんでかって言うと、emacs にはじめて触った日に徹夜して、それから 1 週間の間は確か 3 回しか寝なかったと思う。3 回目の睡眠から目覚めた途端、作り始めてしまった。ちょうど TeX を始めたのと同じころのこと。「動けばいいや」感満載で、今では source code を見る気もしない。)

思想に一貫性がないせいで、実装がゴタゴタしている。

そんな訳で、chibitex から downgrade したその subset, picoTeX をまずはまとめてみたい、と思っています。

その辺の話も、この連載で明らかになるので、今直ちに見なくても支障はないと思いますが.