Advertising TEX

Hans Hagen

I can get upset when I hear TEXies boast about the virtues of TEX compared to for instance Microsoft Word. Not that I feel responsible for defending a program that I never use(d) but attacking something for no good reason makes not much sense to me. It is especially annoying when the attack is accompanied by a presentation that looks pretty bad in design and typography. The best advertisements for TEX should of course come from outside the TEX community, by people impressed by its capabilities. How many TEXies can really claim that Word is bad when they never tried to make something in it with a similar learning curve as they had in TEX or the same amount of energy spent in editing and perfecting a word-processor-made document.

In movies where computer technology plays a role one can encounter weird assumptions about what computers and programs can do. Run into a server room, pull one disk out of a RAID-5 array and get all information from it. Connect some magic device to a usb port of a phone and copy all data from it in seconds. Run a high speed picture or fingerprint scan on a computer (probably on a remote machine) and show all pictures flying by. Okay, it's not so far from other unrealistic aspects in movies, like talking animals, so maybe it is just a metaphor for complexity and speed. When zapping channels on my television I saw figure 1 and as the media box permits replay I could make a picture. I have no clue what the movie was about or what movie it was so a reference is lacking here. Anyway it's interesting that seeing a lot of TEX code flying by can impress someone: the viewer, even if no TEXie will ever see that on the console unless in some error or tracing message and even then it's hard to get that amount. So, the viewer will never realize that what is seen is definitely not what a TEXie wants to see.

So, as that kind of free advertisement doesn't promote TEX well, what of an occasional mentioning of TEX in highly-regarded literature? When reading "From bacteria to Bach and back, the evolution of minds" by Daniel Dennett I ran into the following:

> In Microsoft Word, for instance, there are the typographical operations of superscript and subscript, as illustrated by
>
> $\text{base}^{\text{power}}$
>
> and
>
> $\text{human}_{\text{female}}$
>
> But try to add another superscript to $\text{base}^{\text{power}}$—it *should* work, but it doesn't! In mathematics, you can raise powers to powers to powers forever, but you can't get Microsoft Word to display these (there are other text-editing systems, such as TeX, that can). Now, are we sure that human languages make use of true recursion, or might some or all of them be more like Microsoft Word? Might our interpretation of grammars as recursive be rather an elegant mathematical idealization of the actual "moving parts" of a grammar?

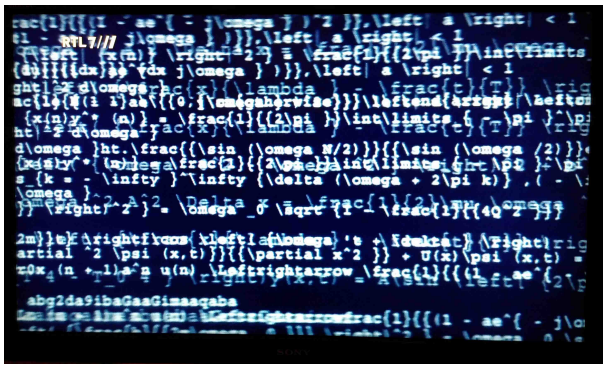

Figure 1: TEX in a movie

Now, that book is a wonderfully interesting read and the author often refers to other sources. When one reads some reference (with a quote) then one assumes that what one reads is correct, and I have no reason to doubt Dennett in this. But this remark about TEX has some curious inaccuracies.[1]

First of all a textual raise or lower is normally not meant to be recursive. Nesting would have interesting consequences for the interline space so one will avoid it whenever possible. There are fonts that have superscript and subscript glyphs and even Unicode has slots for a bunch of characters. I'm not sure what Word does: take the special glyph or use a scaled down copy?

Then there is the reference to TEX where we can accept that the "E" is not lowered but just kept as a regular "e". Actually the mentioning of nested scripts refers to typesetting math and that's what the superscripts and subscripts are for in TEX. In math mode however, one will normally raise or lower symbols and numbers, not words: that happens in text mode.

While Word will use the regular text font when scripting in text mode, a TEX user will either have to use a macro to make sure that the right size (and

[1] Of course one can wonder in general that when one encounters such an inaccuracy, how valid other examples and conclusions are. However, consistency in arguments and confirmation by other sources can help to counter this.

font) is used, or one can revert to math mode. But how to explain that one has to enter math and then explicitly choose the right font? Think of this:

```
efficient\high{efficient} or
efficient$^{\text{efficient}}$ or
\par
{\bf efficient\high{efficient} or
efficient$^{\text{efficient}}$}
```

Which gives (in Cambria)

efficient$^{\text{efficient}}$ or efficient$^{\text{efficient}}$ or
efficient$^{\text{efficient}}$ or efficient$^{\text{efficient}}$

Now this,

```
efficient\high{efficient\high{efficient}} or
efficient$^{\text{efficient$^{\text{efficient}}$}}$ or
\par
{\bf efficient\high{efficient\high{efficient}} or
efficient$^{\text{efficient$^{\text{efficient}}$}}$}
```

will work okay but the math variant is probably quite frightening at a glance for an average Word user (or beginner in TEX) and I can understand why someone would rather stick to click and point.

efficient$^{\text{efficient}^{\text{efficient}}}$ or efficient$^{\text{efficient}^{\text{efficient}}}$ or
efficient$^{\text{efficient}^{\text{efficient}}}$ or efficient$^{\text{efficient}^{\text{efficient}}}$

Oh, and it's tempting to try the following:

```
efficient{\addff{f:superiors}efficient}
```

but that only works with fonts that have such a feature, like Cambria:

efficient$^{\text{efficient}}$

To come back to Dennett's remark: when typesetting math in Word, one just has to switch to the math editing mode and one can have nested scripts! And, when using TEX one should not use math mode for text scripts. So in the end in both systems one has to know what one is doing, and both systems are equally capable.

The recursion example is needed in order to explain how (following recent ideas from Chomsky) for modern humans some recursive mechanism is needed in our wetware. Now, I won't go into details about that (as I can only mess up an excellent explanation) but if you want to refer to TEX in some way, then expansion[2] of (either combined or not) snippets of knowledge might be a more interesting model than recursion, because much of what TEX is capable of relates to expansion. But I leave that to others to explore.[3]

[2] Expanding macros actually works well with tail recursion.

[3] One quickly starts thinking of how `\expandafter`, `\noexpand`, `\unexpanded`, `\protected` and other primitives can be applied to language, understanding and also misunderstanding.

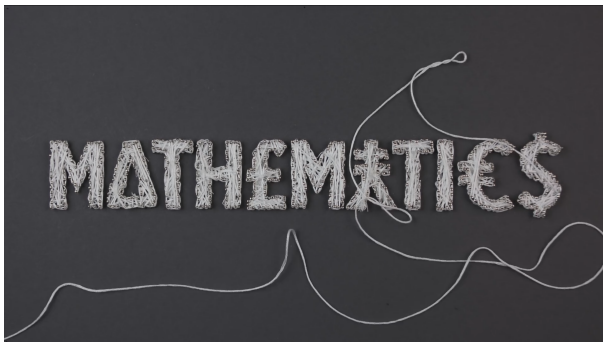

Figure 2: Nicer than TEX

Now, comparing TEX to Word is always kind of tricky: Word is a text editor with typesetting capabilities and TEX is a typesetting engine with programming capabilities. Recursion is not really that relevant in this perspective. Endless recursion in scripts makes little sense and even TEX has its limits there: the TEX math engine only distinguishes three levels (text, script and scriptscript) and sometimes I'd like to have a level more. Deeper nesting is just more of scriptscript unless one explicitly enforces some style. So, it's recursive in the sense that there can be many levels, but it also sort of freezes at level three.

I love TEX and I like what you can do with it and it keeps surprising me. And although mathematics is part of that, I seldom have to typeset math myself. So, I can't help that figure 2 impresses me more. It even has the so-familiar-to-TEXies dollar symbols in it: the poem "Poetry versus Orchestra" written by Hollie McNish, music composed by Jules Buckley and artwork by Martin Pyper (I have the DVD but you can also find it on YouTube). It reminds me of Don Knuth's talk at a TUG meeting. In *TUGboat* 31:2 (2010) you can read Don's announcement of his new typesetting engine iTEX: "Output can be automatically formatted for lasercutters, embroidery machines, 3D printers, milling machines, and other CNC devices . . . ". Now that is something that Word can't do!

◇ Hans Hagen
Pragma ADE
`http://pragma-ade.com`