

How to Count Words in Latex Files?

Adrian Kosmaczewski

2008-03-01

I am a big LaTeX fan, mostly thanks to my friend Cedric who introduced me to it ;) And I don't regret it at all; there is simply no better way to create long, beautiful PDF documents, particularly during these times of dissertation writing! I'm in my last step towards the Master's degree I've been working on for the last two years, and creating documents is an important part of that.

LaTeX works for me, because:

- It's cross-platform (and I need that for my project!);
- It's text based (I can edit the files with any decent editor; personally I use and TexShop and sometimes TextMate);
- I can generate PDF, plain text, RTF, and much more from the same source;
- I can split my documents in several others and work separately in each;
- I can generate meaningful diffs using Subversion (to see what I've changed in every revision);
- I can manage the bibliography for my papers easily (using the awesome BibDesk tool);
- I don't have to cope with a buggy text editor that crashes every so often!
- I can generate gorgeous, absolutely beautiful documents. Easily.

For my last document, the dissertation, I have a numeric limit in the number of words (~ 10K to 15K words) and I need to count the number of words in the documents I generate. Since I'm not using Word, nor KOffice nor OpenOffice, this simple requirement becomes more complex to fulfill. But working in a Unix environment has its benefits; first I found this solution:

```
$ detex file.tex | wc -w
```

This command provides a first approach to the problem; however, it just strips off the LaTeX commands, even those that generate content in the final document. For example, if you have a macro that puts in bold the name of your project, those words will not appear in the final calculation even if they do appear in the final document. Clearly not acceptable. Googling a bit more, I found what I was looking for:

```
$ ps2ascii file.pdf | wc -w
```

In this case we're working on the final PDF document, and of course the final result is much, much more interesting.

Happy typesetting!