inside:

MOTD

by Rob Kolstad

motd

by Rob Kolstad

Rob Kolstad is currently Executive Director of SAGE, the System Administrators Guild. Rob has edited *;login:* for over ten years.

kolstad@sage.org

Leverage

I keep thinking that I will soon be wizened and know everything I need to know. I avoid prejudices and try to judge actions and results more than hype.

I try to partake of technology services so that I can respond intelligently about Web sites, software services, initiatives, ideas, and the like. I pay my bills online. I read my email. I installed a spam filter (what a treat; I am using SpamAssassin and it's so much better than not having it). I have online access throughout my house via strategically placed laptops and other computers. I have both UNIX(-like) and Windows running on various platforms both fixed and portable.

I had the opportunity last week to extend my knowledge of the use of Windows. I had a document that was required to be formatted "nicely" in order that I might distribute it to SAGE members who might wish to see it. For better or worse, the document contained roughly 40 tables and a dozen postscript graphs. Troff is not regarded as the world's best table formatter, so I embarked on implementing something a bit snazzier that would inspire my readers to enjoy the document and its myriad figures.

I had implemented the original document in HTML but found I needed a PDF document. We know that HTML → PDF converters are not renowned for their beauty. I tried to get the document typeset (by someone else) in Quark but initially failed on a quick success due to scheduling and other non-technical reasons. So, I reasoned, Quark can read MSWord files, I'll just format the document in MSWord and then it will be trivially convertable to Quark.

I spent a few minutes writing a short script to extract the document from the HTML so that it could be imported into MSWord and formatted (for the text) and converted to a table (for the tables). After accidentally typing the wrong filename, I learned that MSWord will read HTML directly and do a credible job of laying it out – including tables! I figured my troubles were over.

That was before my ultimate epiphany about MSWord (and, as it turns out, so many of the "tools" under Windows). I found that while MSWord has styles and templates for global formatting conventions, it is difficult to get tables to fall into a consistent format. Each table required personal attention. Even moving the tables to be justified on the right margin was also not possible without clicking on each table and moving it. In fact, I ultimately spent about 25 hours trying to get the document to be passably presentable.

What was the epiphany? Windows has the wrong kind of leverage. I think of leverage as using devices (e.g., computers) or people to increase one's reach or one's ability to solve problems. The Windows tools I was using were insidious. They accepted incremental improvement but never enabled me to improve by an increment of any appreciable size. I could inch closer to my desired solution, but only a little at a time – a snail's pace. "I think I'd like the tables on the right margin," required multiple clicks on each table. Argh!

Why is it insidious? Because it was always possible to move (often to move closer to the solution). At no time did I fail to have some (potentially untried) option or some idea of how to move forward. I have every reason to believe that people who format one table every two months find extraordinary power in the ability to do so with MSWord.

And, to be fair, MSWord works for dozens (hundreds?) of languages. The folks at the international programming contests regularly translate the problems into 65 languages – all using MSWord. This is an incredible feat!

But I hate it. No leverage! More skill or practice only incrementally improves my ability to format tables (in this case). The same seems to be true for any repetitive operation that one would like Windows to perform (e.g., "Please convert these 22 PostScript files to PDF" for our LISA attendees). That's 22 sets of choose-a-file, convert, confirm, and iterate. Ick.

So, I am now familiar with the state of the art. It was an epiphany for me.