

## Stupid Tricks

After the heavy reader response to last April's Forum on "Quirksort", I decided this April to make a fool out of myself instead of you. You'll be back in the line of fire next year. (Be ready.)

I had just returned from twenty-two months of service work in Brazil in 1973 when Richard, an engineer at Lockheed, hired me to paint houses for the summer. It was a side business he had mainly for tax purposes, so he wanted to hire people who needed little supervision. He assumed that Stewart, another young man (age 21, just like me) with similar experience, and I would work well together and turn him a tidy profit.

He was right about Stewart. It seems that I, on the other hand, had a flair for the artistic. I took care to ensure that every line was perfect, that every stroke of the brush blended with its fellows to uniformly cover its target. Clients were pleased and I was in hog heaven. For two weeks, anyway. I was being paid by the hour and just wasn't fast enough, and Richard evidently couldn't afford to fund works of art. He had to "let me go" (a favorite euphemism). I was canned. Sacked. Fired. For painting too well.

I deserved every bit of it. I'm no businessman (obviously), but I know enough to realize that to be profitable, a business must walk the fine line that provides sufficient value while expending only a modest amount of resources. Perfection is not an option. I was overindulging my inclination to "do my best" and refine my personal abilities in a way that Richard's business model didn't allow for.

Unfortunately, too many businesses in our industry follow the same business model, while at the same time it seems too few programmers lust after quality. Surely there must be some common ground between the two parties that allows for excellence.

I once worked for a manager who believed that training should be offered only on rare occasion and only to a precious few as a perk for a job well done. Seems sort of backwards, doesn't it? There need to be resources allocated for "sharpening the saw," to borrow Stephen Covey's term, else programmers and product turn very dull indeed, yet the blind spot induced by the short-term, quarterly-bottom-line mentality of today's management force keeps training, mentoring, and personal development off the priority list. **Stupid management tricks.**

I have also observed programmers who, like Godot's anticipants, take no initiative to expand their competence, preferring instead to be spoon-fed by the next code inspection (if indeed these occur at their shop) or by some training course which, because of their irresponsible attitude, goes in one ear and out the other. **Stupid programmer tricks.**

Over the years I've learned to better balance my yearning for learning with my employer's penchant for profit. Meeting in the middle has required some late-night study, an occasional job change, and regularly making my share of mistakes. Those mistakes, over the years, have included: copying and pasting code instead of factoring it into a separate module, thus using my editor as a bug replication tool; not using assertions; putting real work into assertions (which of course disappear in release mode); returning a reference to a local variable; providing a complex solution when a simpler one would do; and letting myopic managers get under my skin. That's just the Short List, of course. But hey, I'm not as stupid as I used to be. That's something. (But I still paint real slow.)