

## I'm An Idiot...(And Other Lessons From The IT Department)

By Dan Tienes (dan@itoughttobesimple.com)

I'm an idiot.

I'm [stupid](#), [clueless](#), [dumb](#) – hell, I'm a [complete moron](#). I'm so inept, in fact, that a new word has been created to capture my incompetence: “[luser](#).”

I feel terrible about it, I really do; it was never my intention to upset my IT department – heck, the whole IT industry – by not being bright enough to use the wonderful tools they give me. But I just can't seem to get it right.

I mean, I *know* I'm not supposed to click on attachments. Clicking on attachments is bad. My IT department sent me an email explaining this. They were even kind enough to attach a Word document explaining how to set my computer up to prevent the spread of viruses through attachments like...well, like Word documents. I have to admit, that little irony had me scratching my head for a few minutes. Was this some sort of test for us lusers to see if we pay attention? Then I realized the message came from my IT department. And you can't fake an email address. No way.

I think I passed their test.

And yet they still think very little of me. I read their blogs: “[Users are stupid and that needs to be the starting point for software developers](#).” I read their trade magazines: “[No matter how hard we pray...every network is at one time or other exposed to the ultimate technology risk: users](#).” I know, I know, I probably shouldn't be reading these blogs and magazines; it's all highly technical stuff they're talking about, and I'm probably missing the crucial subtext when they refer to me as “[this most dangerous species of wildlife](#).” My problem is that I just don't get it.

Or, not.

The IT profession – [and it's debatable whether IT qualifies as a profession](#) – needs to get its act together and start acting like one. Today, IT behaves more like a high-school clique, knotted together in the cubicle maze, snickering and slandering everyone

---

Copyright © 2007 It Ought To Be Simple Productions

<http://itoughttobesimple.com>

who's not one of them. There's only one other career I know of where blatantly insulting your customers is acceptable, and that's [stand-up comic](#). (Of course being a comic also requires you to be funny, and material like *select \*from users where clue > 0* isn't funny.)

We should be thankful that not all professions are as self-assured and blameless as IT. It would be very demeaning, for example, if doctors – taking a cue from IT – began referring to patients, in public discourse, as “meatbags.” Or if an august publication like [The New England Journal of Medicine](#) published a paper positing that the placebo effect worked because most patients were just idiots who believed anything they were told. I hope I never see a T-shirt for sale to physicians declaring “*You're Not Sick – You're Stupid*” or “*Cancer Patients Are A Waste.*”

That would be a hoot, huh?

“So what,” the IT employee might say. “We're just blowing off steam. So a few less-intelligent-than-me users get their feelings hurt.” Fair enough; I'm sure all professions have some subject that's a dependable butt of jokes. I can't imagine that doctors and lawyers and newspaper reporters don't occasionally vent their spleens about their patients, clients or readers. But if they do, they do it behind the scenes, away from the public eye. For them to do otherwise is to risk public censure (at best) or banishment (at worst). [Doctors can lose their licenses](#), [lawyers can be disbarred](#), and [reporters can be fired for failing to live up to their professional responsibilities](#).

Smarmy IT professionals get magazine columns.

If IT's condescending attitude resulted in nothing more than a few hurt feelings, it wouldn't be such an issue. But the attitude causes much bigger problems: It gives IT a free pass, the right to blame mistakes on users...who are, after all, too stupid to know what they're doing. There's a litany of problems with software: [poor interface design](#), [bugs](#), and [an absurd number of security problems](#). Of course, IT blames sloppy interfaces on computer illiteracy, software bugs on poor user requirements (users just don't know what they want), and security problems on user stupidity.

Consider a common problem: [inconsistent and confusing interfaces](#). What if other businesses placed the blame for inconsistent interfaces on their customers? What if, for example, some Ford cars had the brake pedal on the *right*, and the gas pedal on the *left*, the exact opposite of how it's done today? Worse still, what if the pedals *changed functions while you were driving*? Would automakers say: "Look, we explained the way the pedals worked in the owner's manual. And whenever the pedals are about to change functions, you can feel a tug in the transmission – it's unmistakable. Anyone who knows anything about cars can figure it out. We can't help the fact that drivers are too stupid to get it."

As a user, my ineptitude gives IT grounds to dismiss my feedback and scoff at my complaints. Especially my complaints: *Forget him, he just doesn't get it*. Too often users are the scapegoats for poorly-designed and shoddily built software: "The lusers' deadlines were unrealistic! "The lusers changed the requirements!" or "The lusers forced us to dumb the application down!" And on and on.

It's true all of these things – overambitious deadlines, changing requirements, and design compromises – contribute to failed software projects. But IT doesn't seem to understand (or care) that deadlines and requirements and compromises are all driven by legitimate business needs. No, these things happen because their customers are lusers: hopelessly clueless, moronic, and idiotic.

I worked as a software development director for 10 years. I am guilty of scoffing at my customers. Users made mistakes, and – since I understood the way computer systems really work – the users' mistakes seemed silly. But when I reflected on how those mistakes were made – by ignoring warning messages, for example, that were so numerous and obscure and often irrelevant that clicking "OK...OK...OK" became a habit – it became clear that the problem was not the user, but the godawful software we had saddled them with.

I always viewed the end user as a collaborator and customer, not an obstacle or an idiot. Working with our users helped my team define goals, share in success, and learn

from failures – ours, and theirs. They were the reason we built software; it was our job to make software that made their jobs better. If our work made the user feel like a luser then *we were to blame*. We hadn't done our jobs.

There is, believe it or not, a [code of ethics for software developers](#), developed by the Association for Computer Machinery (ACM). The code spells out the obligations software engineers owe the public. “These obligations are founded in the software engineer’s humanity,” reads the preamble, “in special care owed to people affected by the work of software engineers, and in the unique elements of the practice of software engineering.”

I doubt most IT professionals have ever heard of the code of ethics. In my ten years of experience, it was never once mentioned, nor did I ever come across it in a trade journal or programming magazine. And I certainly never knew of a decision that was influenced by it. IT decisions are instead made based on the urgency of the problem being solved (Is a system down? Is this a pet project of the CEO? Has the deadline already slipped by three months?) or the allure of the technology (Sure, we can do it in a shell script in about an hour, but isn't .NET *cooler*?).

But making a decision governed by a code of ethics? That would be a challenge. IT is objective – either a program works, or it doesn't; either you can ping the network, or you can't. Ambiguity is antithetical. You can't assign a value to “special care.” There's no network protocol that supports it, and no function call that returns it.

So start with the basics: Don't call users idiots. I don't know if that qualifies as “special care,” but it *is* common courtesy. That's a good start.