Release Late, Release Rarely

July 5, 2006

When you look at something you're working on, no matter what it is, you can't help but see past the actual thing to the ideas that inspired it, your plans for extending it, the emotions you've tied to it. But when others look at it, all they see is a piece of junk.

You only get one chance to make a first impression; why have it be "junk"? Once that's associated with your name or project, it's tough to scrape off. Even people who didn't see it themselves may have heard about it second-hand. And once they hear about it, they're not likely to see for themselves. Life's too short to waste it on junk.

But when you release late, after everything has been carefully polished, you can share something of genuine quality. Apple, for example, sometimes releases stupid stuff, but it always looks good. Even when they flub, people give them the benefit of the doubt. "Well, it looks great but I don't really like it" is a lot better then "it's a piece of junk".

Still, you can do better. Releasing means showing it to the world. There's nothing wrong with showing it to friends or experts or even random people in a coffee shop. The friends will give you the emotional support you would have gotten from actual users, without the stress. The experts will point out most of the errors the world would have found, without the insults. And random people will not only give you most of the complaints the public would, they'll also tell you why the public gave up even before bothering to complain.

This is why "release early, release often" works in "open source": you're releasing to a community of insiders. Programmers know what it's like to write programs and they don't mind using things that are unpolished. They can see what you're going to do next and maybe help you get there.

The public isn't like that. Don't treat them like they are.