Puttin' the *Queue* in QA

Edward Grossman, Editor, *ACM Queue*

I'm a total sucker for corny titles, so let me just apologize for the above right off. My intent is not in any way to trivialize the import of quality assurance. Rather, the opposite: QA is *so* important that the *ACM Queue* Editorial Advisory Board felt it merited its own dedicated special report.

Unlike many of the topics *Queue* tackles, QA isn't sexy or exciting. It isn't new, hip, or happenin'. You'll not see many magazine covers proclaiming QA to be the next big thing, in the vein of past cover crazes: Push Technology Will Change the World!; or XML, XML, XML!; or Java, This Changes Everything!

While QA isn't any of these things, it is clearly one thing—important. Very important.

QA is hard, QA is contentious. It's easy to do poorly, and, sadly, all too easy to not *do* at all. I'm reminded of my Web scripting days, and I confess, I am guilty. Too often schedules and “good enough” ruled the day, and QA consisted of an overstuffed e-mail box:

support@thesitecrashedagain.com.

Of course, this raises an interesting philosophical question: from a strictly utilitarian perspective, is QA, in excess of “good enough” (for various values of “good enough”), a waste of money? Whatever the merits of “quality for quality's sake,” QA is arguably something else.

QA is (more or less) a process by which a product's actual function is compared with its intended function—measured against the spec, presuming you have one. Does this software do what it's supposed to? And, of course, that spectrum of what something is supposed to do can vary widely—“always stop drill spinning within 0.01 seconds” vs. “give most users the document they were looking for without making them too frustrated.”

Therein lies my problem—I am not a utilitarian. At some level I care about quality for quality's sake, quality above and beyond its simple utility. That's not uncommon; I'm sure many of you share that sentiment. But combine it with my tendency to be lazy, and “quality” becomes something of a soft and squishy issue (not very QA!). I want the product to be good, for my current mental idea of “good” (which is calculated something

Actual function

VS. INTENDED

FUNCTION

along the lines of: good
= how_scary_my_boss_is
– [time_from_last_meal +
time_to_next_meal]... I'm
kidding!).

This is all a long (and somewhat rambling) way of saying it's a good thing I'm an editor and not a programmer. And, it's a good thing *Queue's* content is generated by our Editorial Advisory Board, not me. *Queue's* special report on QA is not what journalists and editors think of QA, but QA from a technologist's perspective. The articles in this issue are written by people who *do* QA, who live and breathe QA. I think you'll find this unsexy topic treated with the import it deserves, and hopefully you'll learn a thing or two. I know I did.

One last thing before letting you go. I've received such great feedback from readers these past few months (thank you Meditation Code revelators and Googlers of Burdian's Ass—see Letters on page 10 to see who won the *Queue* mug!) that I'd like to invite some more. I've got two questions. First, I'd like to know how many of you out there have a formal vs. informal QA process (e.g., separate QA people/process vs. we're really careful); please drop me a note—I'll keep your answers anonymous—and I'll present the results in this column next month.

Second, as part of our own QA, I'd love to get your thoughts on how *Queue* is doing against its spec: Are we functioning as intended? Or maybe that's better phrased: Are we providing you with the content you expect? Are we educating and interesting you? Please be specific, and be honest—what articles (in this issue or earlier) just didn't cut it? Which were great (hopefully at least some!)? Your answers will help *Queue* be the best it can be, and to be the truly community-driven publication it's set out to be (am I gettin' corny again?). Enjoy! ☺

EDWARD GROSSMAN is responsible for *Queue*, so blame him if you don't like it: edgrossman@acmqueue.com. In earlier incarnations he was a development project manager at a still-in-business dot-com and a closet coder (his parents still don't know—“Our son Ed? Oy, he works with computers, doing something”).