Responses to “Software Engineering: An Idea Whose Time Has Come and Gone?”

We’ve received numerous letters regarding Tom DeMarco’s Viewpoints essay in the July/August 2009 issue. Here are short summaries of some of the letters we’ve received thus far. The complete text of the letters are posted at www.computer.org/software/demarco. The page also has instructions on how to add your comments to the discussion. —Hakan Erdogmus, editor in chief

Andrew Raybould of Thales Fund Management writes:
Like Tom, I once had high hopes for software engineering. … Defining a problem to exclude the difficult cases is the opposite of solving them. … I’m willing to accept that controlled procedures are a necessary, though never sufficient, condition for success. I think there may be a way forward for software engineering, but it will not be as yesterday’s discipline.

Bob McCann of Lockheed Martin Aeronautics writes:
What if software projects were incentivized like Silicon Valley startups? Motivation is the key ingredient that software engineering as currently constituted does not address. … It’s possible to measure the ROI of effective metrics. Without metrics, it’s impossible to counter mistaken beliefs distinguishing among good, excellent, and moderately poor performance.

Capers Jones of Software Productivity Research writes:
As always, Tom makes good points. But if risk analysis and value analysis are added to the discussion, the conclusions are the opposite of Tom’s. … The more valuable the software, the more quality and change control are needed. … Poor quality control, poor change control, inadequate estimates, and poor status tracking are the main reasons that software projects fail.

Tom Gilb of Result Planning Ltd. writes:
Tom is saying something very important, but so softly that many might miss it. Let me rephrase:
• We need to use the best means we can find, even software engineering—revisited—to shift our focus away from narrow control over project time and cost alone.

Kannan Vijayaraghavan writes:
DeMarco’s willingness to look at the past, reflect, and adjust past views from the perspective of accumulated experience increases his stature in my eyes. … If we agree that software engineers are self-driven, imaginative people, we should give them goals rather than tell them what to do and how to do it. Indeed, as my daughter enters her teens, this is precisely what I might do.

R.T. Sakthidaran of the KLN College of Engineering writes:
Allow dissenting opinions in publications even if they do not come from eminent personalities. … If we cannot monetize an intangible benefit, let’s not discard it as useless. … We, and every member of IEEE, must appreciate Tom’s courage and honesty to express his reflections. I hope more and more engineers come forward and honestly review their life’s contributions.

Luke Nguyen and Alvin Teh write:
Without control, the likelihood and impact of risks are unknown and therefore an effective risk mitigation strategy is nigh impossible. … ROI is not limited to money. It can include scientific advancement (Mars Lander), better defense (Joint Strike Fighter), knowledge accessibility (Google Scholar), and more.

Erratum

In the July/Aug. 2009 issue, in “Commodification of Industrial Software: A Case for Open Source” (pp. 77–83), the first paragraph of the section “Philips, Agfa, and the DICOM Validation Toolset” misstated Dr. David Clunie’s role in the standardization of medical imaging. The paragraph should be the following:

“Medical imaging for diagnostic purposes has been subject to standards since the end of the 1980s. Over time, radiologists and surgeons needed to interchange images made on equipment from different manufacturers. This led to an effort toward standardization of medical-image interchange, which resulted in the Digital Imaging and Communications in Medicine Standard (DICOM, http://medical.nema.org).”

We apologize for the error.