

## Clueless—and Oblivious

**Warren Harrison**

**M**any of us have heard someone say, upon completing a course of study, “Now I know exactly how much I don’t know.” Even Thomas Jefferson is said to have observed that “he who knows best knows how little he knows.” I’ve seen this phenomenon displayed by both individual programmers and entire software development organizations throughout my career in the software industry.



Invariably when I visit an organization and am told that they have the “software problem” under control, I can count on both an uphill battle over making any meaningful changes and a good dose of grouching over “naive customers” and “unreliable contractors.” Conversely, I’m often pleasantly surprised when I encounter an organization that seems almost embarrassed to explain their software development process to me—they’ll often say this is how they currently do things but hasten to add that they’re trying to improve one aspect or another. More often than not, while they might not be perfect, they’re better than the vast majority of professional software development organizations out there. Likewise, we’ve probably all run into hot-shot programmers who speak in bits and bytes, promise the moon, but somehow never seem to deliver their work products (at least ones that actually work) on time.

I never really thought of this as some kind of universal pattern. Rather, I simply assumed that it was the luck of the draw, and I continued to

take both organizational as well as personal self-assessments of capabilities at face value. After all, we’re taught that self-confidence is important, and to show doubt in your capabilities is the equivalent of hanging a sign that says “INCOMPETENT” around your neck.

### **Unskilled and unaware**

My past assumptions now seem to be incorrect. A friend recently brought a paper published in a well-known psychology journal to my attention: “Unskilled and Unaware of It: How Difficulties in Recognizing One’s Own Incompetence Lead to Inflated Self-Assessments” (*Journal of Personality and Social Psychology*, Dec. 1999, pp. 1121–1134; [www.apa.org/journals/psp/psp7761121.html](http://www.apa.org/journals/psp/psp7761121.html)). This article, by Justin Kruger and David Dunning, is fast becoming a classic in the field of psychology. It’s also brought all my past experiences into crystal-clear focus and provided me with a deeper understanding than ever before of why some people and organizations just never seem to “get it.”

Kruger and Dunning point out in their paper:

*We argue that when people are incompetent in the strategies they adopt to achieve success and satisfaction, they suffer a dual burden: Not only do they reach erroneous conclusions and make unfortunate choices, but their incompetence robs them of the ability to realize it. Instead, ... , they are left with the mistaken impression that they are doing just fine.*

In other words, people (and organizations)

### DEPARTMENT EDITORS

**Bookshelf:** Warren Keuffel,  
wkeuffel@computer.org

**Construction:** Andy Hunt and Dave Thomas,  
(andy, dave)@pragmaticprogrammer.com

**Design:** Martin Fowler,  
fowler@acm.org

**Loyal Opposition:** Robert Glass,  
rglass@indiana.edu

**Manager:** Don Reifer,  
d.reifer@ieee.org

**Quality Time:** Nancy Eickelmann,  
nancy.eickelmann@motorola.com,  
and Jane Hayes, hayes@cs.uky.edu

**Requirements:** Suzanne Robertson,  
suzanne@systemsguild.com

### STAFF EDITORS

Senior Lead Editor  
**Dale C. Strok**  
dstrok@computer.org

Group Managing Editor  
**Crystal Shif**

Senior Editors  
**Shani Murray and Dennis Taylor**

Staff Editor  
**Rita Scanlan**

Assistant Editor  
**Rebecca Deuel**

Magazine Assistant  
**Hilda Hosillos**, software@computer.org

Art Director  
**Toni Van Buskirk**

Cover Illustration      Technical Illustrator  
**Dirk Hagner**              **Alex Torres**

Production Assistant      Production Artist  
**Monette Velasco**        **Carmen Flores-Garvey**

Executive Director  
**David Hennage**

Publisher                      Assistant Publisher  
**Angela Burgess**        **Dick Price**

Membership/Circulation Marketing Manager  
**Georgann Carter**

Business Development Manager  
**Sandra Brown**

Senior Production Coordinator  
**Marian Anderson**

### CONTRIBUTING EDITORS

Robert Glass, Thomas Centrella,  
Kirk Kroeker, Keri Schreiner

**Editorial:** All submissions are subject to editing for clarity, style, and space. Unless otherwise stated, bylined articles and departments, as well as product and service descriptions, reflect the author's or firm's opinion. Inclusion in *IEEE Software* does not necessarily constitute endorsement by the IEEE or the IEEE Computer Society.

**To Submit:** Access the IEEE Computer Society's Web-based system, Manuscript Central, at <http://cs-ieee.manuscriptcentral.com/index.html>. Be sure to select the right manuscript type when submitting. Articles must be original and not exceed 5,400 words including figures and tables, which count for 200 words each.

that aren't too good at what they do often don't realize it by virtue of the fact that they aren't too good at what they do.

The paper presents a series of studies that analyzed participants' self-assessments of their appreciation of humor, their ability to apply logical reasoning, and their understanding of English grammar. In short, they found that incompetent participants rated their relative performance higher than was really the case, while their more skilled colleagues tended to underestimate their relative performance, if for no other reason than believing everyone else did as well as they did.

As we'd expect, the researchers also found that with training, the originally incompetent participants' reassessment of their performance was closer to their real performance—in other words, not only had their performance improved, so had their ability to predict their performance. Importantly, their ability to predict their performance didn't necessarily arise from actually performing better, but perhaps simply because of increased knowledge.

### Software development and psychology

This is all fine and good, but what does a psychological study have to do with developing software? Kruger and Dunning point out that “incompetent individuals, compared with their more competent peers, will dramatically overestimate their ability and performance relative to objective criteria.”

Interestingly enough, we find that as software development organizations become more mature in terms of their CMM levels—for instance, moving from Level 2 to Level 3—the frequency of budget and schedule overruns decreases. For example, Patricia Lawlis, Robert Flowe, and James Thordahl compared the deviation between planned and actual costs for 31 projects performed by a variety of companies at CMM Levels 1, 2, and 3 (“A Correlational Study of the CMM and Software Development Performance,” *Crosstalk*, Sept. 1995, [www.stsc.hill.af.mil/crosstalk/1995/09/Correlat.asp](http://www.stsc.hill.af.mil/crosstalk/1995/09/Correlat.asp)).

Their results suggest that “as an organization matures from Level 1 to Level 5 ... variability of the actual results about the target decreases, i.e., performance becomes more predictable.”

The elimination of frequent cost overruns is sometimes explained by pointing to increased productivity and new efficiencies as an organization's process matures. Yet another explanation is that by imposing phases, documentation, and other artifacts of “increased maturity” (at least in the world of CMM), projects are simply more predictable. After all, it's easier to come up with a more accurate cost estimate for a product whose behavior is completely specified than for a product whose behavior isn't specified at all.

However, the Kruger and Dunning paper provides an alternate explanation that bears some thought. More mature organizations might avoid overruns for reasons other than greater productivity or efficiencies, or even imposition of additional process steps or documentation. Rather, they might be eliminating overruns because, in the process of increasing their maturity, these organizations simply became more aware of what they know.

### Implications for software development

If it were true that the important thing is gaining knowledge, the implications would be tremendous. Rather than actually changing their processes, organizations could simply study their processes to gain more knowledge. This doesn't mean that an organization wouldn't change their process if it made sense, but it does imply that it isn't simply changing the process but rather the deeper understanding of how their organization works that yields the gain.

This is consistent with a variety of process improvement movements. For instance, Watts Humphrey's Personal Software Process has programmers begin by recording how they spend their time. This self-recorded information plays a critical role in the PSP framework. Obviously, programmers gain a better understanding of what they do and how they

## Thanks to Don Reifer

For the last four years, Don Reifer's Manager column has added value to *IEEE Software*'s pages. His controversial and thought-provoking articles have helped managers increase their understanding of the ideas and technologies that lead to successful projects. Sadly, this is the last issue carrying Don's column. Members of the editorial board, including the editor in chief and most of the columnists, are appointed to two-year terms, with the possibility of one renewal. Don is now at the end of his second term.

While I was preparing this farewell, I decided to take a look at Don's past columns. Interestingly enough, his very first column mentioned collecting metrics. Consequently, I decided to collect a few metrics of my own. Over the past 26 issues, Don's column has contributed 79 pages of content to the magazine, which is close to the size of an entire issue. Also, as Don promised in his first column, he enriched the magazine by encouraging guest columnists so that we could get the perspectives of other leaders in the field. In fact, 16 of his 26 columns were written by guests such as Barry Boehm, Dick Fairley, Watts Humphrey, Walker Royce, and others.

Don has had a tremendous impact on the magazine, and his column will be missed greatly. On behalf of the entire *IEEE Software* family, I'd like to thank Don for all his contributions and wish him the very best of luck in the future.

do it if they actually track how they're spending their time.

Likewise, we're admonished to begin every process improvement effort with a baseline measurement. If my conjecture is true, perhaps we can gain much of the process improvement benefit (at least with regard to schedules and predictability) by stopping there.

I want to make clear that I'm not advocating the wholesale elimination of your process improvement efforts. However, a great deal may stand to be gained by examining the relative value

of different aspects of your process improvement efforts.

## What do you think?

I'd like to hear your thoughts on this issue. Is it possible to gain the benefits of process improvement without really changing the process? What's the value of knowledge? Does anyone have any good stories about clueless organizations or colleagues?

Please write me at warren.harrison@computer.org.

I'm doing just fine, thank you. ☺

## Correction and Apology

In the last issue of *IEEE Software*, Melody Moore's column From Your Technical Council provided an update on the Technical Council on Software Engineering's Standards Committee. In this column, Melody used a common question-and-answer format, in which she'd pose a question and then, on the basis of notes she'd taken during an interview with committee chair Paul Croll, paraphrase Paul's answers.

Unfortunately, the "answers" could be construed to be direct quotes from Mr. Croll and contained a number of errors. Melody and *IEEE Software*'s editors apologize to Mr. Croll and our readers for any confusion or inconvenience that might have occurred. To correct matters to the best of our ability, we are publishing a corrected column on the inside back cover of this issue. This gives Mr. Croll an opportunity to answer these questions in his own words and set the record straight with regard to the TCSE's Standards Committee activities.

## EDITOR IN CHIEF

### Warren Harrison

10662 Los Vaqueros Circle  
Los Alamitos, CA 90720-1314  
warren.harrison@computer.org

EDITOR IN CHIEF EMERITUS:  
Steve McConnell, Construx Software  
stevemcc@construx.com

## ASSOCIATE EDITORS IN CHIEF

**Education and Training:** Don Bagert, Rose-Hulman Inst. of Technology; don.bagert@rose-hulman.edu

**Design:** Philippe Kruchten, University of British Columbia; kruchten@ieeeg.org

**Requirements:** Christof Ebert, Alcatel; christof.ebert@alcatel.com

**Management:** Ann Miller, University of Missouri, Rolla; millera@ece.umar.edu

**Quality:** Stan Rifkin, Master Systems; sr@master-systems.com

**Experience Reports:** Wolfgang Strigel, Software Productivity Center; strigel@spc.ca

## EDITORIAL BOARD

Nancy Eickelmann, Motorola Labs  
Richard Fairley, OGI School of Science & Engineering  
Martin Fowler, ThoughtWorks  
Jane Hayes, University of Kentucky  
Andy Hunt, Pragmatic Programmers  
Warren Keuffel, independent consultant  
Karen Mackey, Cisco Systems  
Deependra Moitra, Infosys Technologies, India  
Don Reifer, Reifer Consultants  
Suzanne Robertson, Atlantic Systems Guild  
Richard H. Thayer, Calif. State Univ. Sacramento  
Dave Thomas, Pragmatic Programmers

## INDUSTRY ADVISORY BOARD

Robert Cochran, Catalyst Software (chair)  
Dave Aucsmith, Microsoft  
Maarten Boasson, Quaerendo Invenietis  
Annie Kuntzmann-Combelles, Q-Labs  
David Dorenbos, Motorola Labs  
Enrique Draier, MAPA LatinAmerica  
Dehua Ju, ASTI Shanghai  
Tomoo Matsubara, Matsubara Consulting  
Dorothy McKinney, Lockheed Martin Space Systems  
Stephen Mellor, Project Technology  
Susan Mickel, Lockheed Martin  
Dave Moore, Vulcan Northwest  
Melissa Murphy, Sandia National Laboratories  
Grant Rule, Software Measurement Services  
Girish Seshagiri, Advanced Information Services  
Martyn Thomas, Praxis  
John Vu, The Boeing Company  
Simon Wright, Integrated Chipware  
Jeffrey Voas, Digital

## MAGAZINE OPERATIONS COMMITTEE

Bill Schilit (chair), Jean Bacon, Pradip Bose, Doris L. Carver, George Cybenko, John C. Dill, Frank E. Ferrante, Robert E. Filman, Forouzan Golshani, David Alan Grier, Rajesh Gupta, Warren Harrison, Mahadev Satyanarayanan, Nigel Shadbolt, Francis Sullivan

## PUBLICATIONS BOARD

Michael R. Williams (chair), Michael Blaha, Mark Christensen, Sorel Reisman, John Rokne, Bill Schilit, Linda Shafer, Steven L. Tanimoto, Anand Tripathi