Basic Etiquette of Technical Communication

Diomidis Spinellis

Parents spend years trying to teach their children to be polite, and some of us had to learn at school how to properly address an archbishop. Yet, it seems that advice on courteousness and politeness in technical communication is in short supply; most of us learn these skills through what’s euphemistically called “on the job training.” With enough bruises on my back to demonstrate the amount and variety of my experience in this area (though not my skill), here are some of the things I’ve learned.

Talking to Humans

We developers spend most of our time issuing instructions for computers to execute. This type of command-oriented work can easily lead to déformation professionnelle (see also J. Bigler’s alternative interpretation at www.mit.edu/~jcb/tact.html); I can still remember, years ago, a Navy officer who was talking to his son as if he was ordering a sailor. When we compose a mail message or open a chat window, our keystrokes are directed to another human, not to a shell’s command-line interface. Therefore, we should switch our tone to courteousness, kindness, and consideration. “Please” and “thank you” aren’t part of SQL (or even Cobol; but interestingly “please” is an important part of Intercal), but they should be sprinkled liberally in every discussion between humans. Are you asking a colleague to do something for you at the end of the business day? This isn’t a batch job that a computer will run in the background. Think of how your request may affect your colleague’s family life. Ask him whether he can do it without too much hardship, and at the very least apologize for the urgency of your request.

Starting your exchange with some (sincere) flattery can work wonders. This is especially important if harsh criticism is to follow; it will help you express yourself in a more compassionate way and lift the spirits of the unfortunate soul who will read your words. Imagine the feelings of your email’s recipient by reading your message again through his eyes; according to human-communication theory, he will interpret the email more negatively than it was intended. Therefore, aim to encourage rather than complain. If your email is especially harsh, don’t send it immediately. Put it aside and sleep on it or ask other, more experienced colleagues for advice. Although Google is experimenting with a feature that lets you revoke an email within a very small grace period, in general there’s no way to undo a sent message—you can only regret the damage it made.

In technical discussions, focus on technology issues, not personal weaknesses. Read the message, “You indent with horrible inconsistency like a loser” as “the code in Foo.java can be better indented”—this has to do with the code, not you. Similarly, instead of shouting, “Your choices of method names for the class Foo are awful,” phrase your concern as, “The methods of class Foo would be easier to remember if they were verbs.” More concretely, Linda Rising, author of Design Patterns in Communications, recommends the following format: “Deliver an Oreo cookie by saying something nice, then present your suggestion of improvement, then close with an appreciation.”

Email Smarts

Every email should tackle one topic and that topic...
Every email should tackle one topic and that topic should be the subject line.

Although a code review with an archbishop is unlikely, you’ll sometimes communicate with your organization’s big shots. There’s no need to be servile in such situations. If you’re sincere, avoid technical jargon, and appreciate the priorities and constraints of the higher ups, you’ll do fine. Remember! You can’t go wrong when you’re considerate, polite, and respectful to everyone.

Diomidis Spinellis is a professor in the Department of Management Science and Technology at the Athens University of Economics and Business and the author of Code Quality: The Open Source Perspective (Addison-Wesley, 2006). Contact him at dds@uueb.gr.

Post your comments online by visiting the column’s blog: www.spinellis.gr/tools