

The Critical Voice

by [Frank Liechtenstein](#)

Are you one of those people who fail to appreciate that XML will revolutionize computing because it represents the ultimate convergence of all information in one data format? Are you too ignorant to see that Java is the future of programming because, in a language where there are no pointers, there can be no null pointer exceptions? Then this is for you! *The Critical Voice* presents the irrefutable truths held by technology managers, sales representatives, journalists, and Hollywood screenwriters from a purely scientific point of view.

This time, we look at the popular conception that any computation, even if it takes *forever* to do by hand, can be done using computers in *no* time. No, seriously; in *no* time. Let's call a computer *supercool* to a degree d if it takes time $1/d$ measured in 18-month periods to execute an operation. According to Moore's law, computers double in their supercoolness every 18 months. So the supercoolness of computers after $18t$ months is $d(t) = 2^t$. This means that by the time any operation would finish on any given computer, computers will already be available which will have executed the operation in less time!

Say we are at point in time $t = 0$; then our computers are supercool to a degree $2^0 = 1$, so one operation takes $18/1 = 18$ months to finish. But after 18 months, computers will, according to Moore's law, already be twice as supercool. So $d = 2$, and it takes only $18/2 = 9$ months to finish. But after 9 months, computers are 1.42 times as supercool as they used to be, so it really only takes $9/1.42 = 6.33$ months, and so on.

You fail to see the irrefutable truths in that argument? Well, that's your fault. I used math, so it must be right. And, furthermore, since we have just shown that no operation takes any time, we can also conclude that $P = NP$, although I never quite

understood what was so difficult about that in the first place. How about $N = 1$, huh? This really goes to show that, yes, you *do* need math to understand computer science, and if the crowd of gray-haired old men in theoretical computer science departments practiced what they preached, they really wouldn't have to waste their time on that one.

No, if they were actually concerned about coming up with a theory of computability that mattered, they would invest some time trying to characterize the class of problems that remain undecided because, according to Murphy's law of Windows programming, after finishing 98% of a decision procedure, the system gives you a bluescreen. I already have an approach to this one: the class of problems that are Murphy-decidable equals the class of problems that can be decided before finishing 98% of the decision procedure. I'm not quite sure how to formalize this strictly, but I'm quite confident that Moore's law supports a statement of the form that, every 18 months, the probability doubles that a time machine becomes available, which could then be used to send the result of the computation back in time to before the bluescreen appeared, using an XML format.

You fail to see how that proposal deserves a research grant? Well, again, that's your fault. I proposed to use XML, so it must be worth something. If you're clever, you'll do the same thing, and we can discuss my time machine at the *5th Joint ACM/IEEE Symposium on Weird XML-Related Stuff* which goes down in Hawaii this year.

This finally brings us to the point where I can conclude this article by shamelessly repeating my introduction. The Critical Voice *has presented* the irrefutable truths held by technology managers, sales representatives, journalists, and Hollywood screenwriters from a purely scientific point of view. So there.

Biography

Frank Liechtenstein received his MSc in Applied Information Technology at the Andreas Hofer University Innsbruck and is currently pursuing a PhD in XML-related studies at the International Institute of Technology in Brussels.