The Dao of Microsoft: Constraining Assumptions & Semantic Rigidity

by M. E. Kabay, PhD, CISSP-ISSMP
CTO, School of Graduate Studies
Norwich University, Northfield VT

The day before I began writing this article, I stopped in at the coffee shop across the street from the School of Graduate Studies in bustling downtown Northfield, Vermont (about which one of our vice presidents says warningly, “Well, we have some pretty rough traffic jams during rush minute”). I looked at a tempting brownie (the small cake kind, not the young girl kind, you evil-minded readers) and asked the youngster behind the counter, “Does that have peanuts?” She answered promptly, “Yes!” As I was putting the brownie back regretfully, an older clerk spoke up: “Actually, those are almonds and hazelnuts, not peanuts.” “Ah,” I said, and promptly bought the brownie. The youngster apologized, saying, “Oh sorry, I assumed you were allergic to peanuts.” “No,” I replied, “I just detest the taste of peanuts with chocolate.”

This incident came to mind as I was thinking about a problem in Outlook 2007. As Doug VanBenthuyens pointed out in July 2006 <http://blogs.3sharp.com/Blog/dougv/archive/2006/07/28/1630.aspx>, older versions of Outlook have long allowed users to store all kinds of text as “signatures,” effectively serving as keyboard macros. For example, I have dozens of text strings including a long signature, a short signature, a letter of thanks to readers, an explanation of how to spell and pronounce my name, and so on. VanBenthuyens noted, “Unfortunately, it no longer seems possible to insert multiple signatures in an e-mail without adding steps (like copy/paste). As expected, Signatures get their own place on a ribbon (Message | Include | Signature). The problem is, when you choose one signature, the one that was already in the e-mail disappears.” Worse yet, sometimes part or even all of the e-mail message disappears with the old signature.

The Microsoft engineers’ errors, in my opinion, were three: they made unwarranted assumptions, they exercised semantic rigidity, and they deprived the user of reasonable control.

I’ve been programming computers since 1965 and teaching programming since 1977. One of the lessons I teach my systems engineering students is to be careful about limiting the power of users without having a good reason for the limitation. In this case, Microsoft engineers presumably assumed that it was impossible for anyone to want to have two signatures in one document. Even if we limit our discussion to signatures for the moment, that assumption seems silly to me; for example, it might be perfectly reasonable to store a short signature (e.g., “Best wishes,” name, title, phone number) and also store a block of details (additional phone numbers, website URL, and so on) to add to that short signature under certain circumstances.

Second, the engineers seem to have been so influenced by the label “signatures” that they discounted any other possible use of the feature. Granted, the Office 2007 suite has other ways of storing keyboard macros. For example, one can store relatively short strings in the AutoCorrect list and use an unusual keystroke sequence (e.g., “=s=”) as a substitute for a particular string. Another way of storing any kind of block of text is the Building Blocks Organizer. I use this feature all the time when I am editing student papers to insert standard suggestions on word usage or grammar. Nonetheless, there is no harm in allowing signatures to be anything the user
wants.

Third, there is no option available to override the engineer’s decision to suppress previous signatures when adding a new one to an e-mail message. You would think that the proliferation of checkboxes for all manner of options in Outlook 2007 and other Office 2007 programs establishes the principle that user control is good; why this particular limitation should be forced upon users is a mystery to me.

Before I close, I want to address a recurring problem I face as I write my little homilies. “What,” some readers demand in exasperation, “does this have to do with security?” Well, I take a very broad view of security that includes Donn Parker’s concept of utility or usefulness as an essential attribute of information. The example I have dissected today is an illustration of the damage to utility that unfounded assumptions can wreak on a system. I hope that readers will apply the principles demonstrated in my analysis to their own work as they design software, networks and policies.

* * *

M. E. Kabay, PhD, CISSP-ISSMP is Program Director of the Master of Science in Information Assurance <http://www.graduate.norwich.edu/infoassurance/> and CTO of the School of Graduate Studies at Norwich University in Northfield, VT. Mich can be reached by e-mail at <mailto:mekabay@gmail.com>; Web site at <http://www.mekabay.com/index.htm>.

Copyright © 2007 M. E. Kabay. All rights reserved.

Permission is hereby granted to *Network World* to distribute this article at will, to post it without limit on any Web site, and to republish it in any way they see fit.