'KILL THAT CODE!'

A casual attitude toward program maintenance can have some deadly results for the organization.

By
Gerald M. Weinberg
founder
Weinberg & Weinberg
Lincoln, NE

My wife, Dani, is an anthropologist by profession, but one of her hobbies is training dogs. The combination of the two produces some interesting ideas. The other day she described how to train attack dogs to keep them from being dangerous. As usual, the big problem with attack dogs is not the dogs, but the people.

When someone hears that a dog is attack trained, chances are about one in three that they'll turn to the dog and as a joke or just to see what the dog will do command: "Kill." To protect against this idiotic human behavior, trainers never use words like "kill" as the attack command. Instead, they use innocent words like "breathe" that would never be given in jest in a command voice.

This protection is needed because a trained dog is an information processing machine in some ways much like a computer. A single arbitrary command could mean anything to a dog, depending on how it was trained—or programmed. This arbitrariness doesn't matter much if it's not an attack dog. The owner may be embarrassed when Rover heels on the command to stay, but nothing much is lost. If the dog is trained to go for the throat, it's an entirely different matter.

It's the same with computers. Because they are programmed and many words in programs have arbitrary meanings, a single mistake can turn a helpful computer into one that can attack and kill an entire enterprise.

That's why I've never understood why management takes such a casual attitude toward program maintenance. Time and again, I hear managers explain that maintenance can be done by less intelligent people operating without all the formal controls of development because it's not very critical. And no amount of argument seems able to convince them differently. Until they have a costly maintenance blunder.

Most expensive errors

Fortunately, costly maintenance blunders are rather common. Managers are learning fast even though the tuition is high. I keep a confidential list of the world's most expensive programming errors. All of the top ten on the current list are maintenance errors. The top three cost their organizations $1,600,000,000, $900,000,000, and $245,000,000—and each one involved the change of a *single digit* in a previously correct program.

In all three cases, the change was "so trivial" it was instituted casually. A supervisor told a low-level maintenance programmer to change that digit without any written instructions, no test plan or nobody to read over the change. Indeed, no controls whatsoever existed between that one programmer and the day-to-day operations of the organization. It was exactly like having an attack dog trained to respond to "Kill," or perhaps to "Hello."

I've done some studies, confirmed by others, about the chances of a maintenance change being done incorrectly depending on the size of the change. Here's the first part of the table:

Lines changed	Chance of error
1	50%
2	60%
3	65%
4	70%
5	75%

Development programmers are often shocked for two reasons to see this high rate. In the first place, development changes are simpler. They involve cleaner, smaller, better structured code that has not been changed many times before. It does not have unexpected linkages. Such linkages were involved in each of my top three disasters.

Secondly, the consequences of an erroneous change during development are smaller. The error can be corrected without affecting real operations. Thus, development programmers don't take that much notice of their errors. They tend to underestimate their frequency.

In development, you simply fix errors and go on your merry way. Not so in maintenance where you must mop up the damage the error causes and spend countless hours in meetings explaining why it will never happen again—until the next time.

SOFTWARE/expo

Jerry Weinberg will deliver the keynote address at the Software/expo National conference in Chicago next month. He has over 25 years of experience as a consultant and researcher and is recognized as an accomplished teacher, lecturer and author. Among his several books are **The Psychology of Computer Programming** ***and*** **An Introduction to General Systems Thinking.**

Ignorant programmers

For these two reasons, development programmers interpret these frequent errors as indicative of the ignorance or inexperience of maintenance programmers. But if we continue the table down a few lines, it shows the cause cannot be either ignorance or inexperience.

Lines changed	Chance of error
10	50%
20	35%

The decrease in error rate as the size of change increases shows maintenance programmers perfectly capable of better work than their record with small changes indicates. My explanation of the higher rates for small changes is that they are not taken seriously. They are done carelessly and without controls. How many times have you heard a programmer say: "No problem! All I have to do is change one line!"?

And how many times have you heard their managers agree with them? Or even encourage them to work "quick and dirty" when it's only a small change?

That attitude would be sensible if "small" changes were truly small; if "maintenance" of a program were actually like maintenance of an apartment building. The janitor can change one washer in the kitchen sink without great risk of causing the building to collapse and bury all of its occupants. It's not safe to make the same assumption for a production program. But because we are so free and arbitrary with words, the word "maintenance" has been misappropriated from the one circumstance to the other.

Careless and unthinking

Whoever coined the term "maintenance" for computer programs was as careless and unthinking as the person who trains a dog to attack on the command "Kill."

With the wisdom of hindsight, I suggest that the "maintenance" programmer is more like a brain surgeon than a janitor. Opening up a working system is more like opening up a human brain and replacing a nerve than opening up a sink and replacing a washer. Would maintenance be easier to manage if it was called "Software Brain Surgery?"

Think about it this way. Suppose you had a bad habit like saying "Kill" to attack dogs. Would you go to a brain surgeon and say, "Just open up my skull, Doc, and remove that one little habit. And please do a quick and dirty job it's only a small change. Just a little maintenance job."

For an extra copy of this article, please enter No. 23 on the Reader Information Card.