

Letters

A bug by any other name

"Warning: This Software May Not Be Safe" (SN:9/13/86,p.171) brings to mind the time before computers and the language of the "hardware-software" dichotomy. Complex telephone systems, operated by means of relays and switches, had built-in software; the rules by which the system performed were hard-wired by design. In parallel to the Bank of New York debacle that resulted when an overload was presented, AT&T had a glitch in its No. 1 Crossbar System that took seven years to unearth. It appeared during overloads, causing a total collapse of the system in handling both originating and terminating calls. When a severe ice storm hit Washington, D.C., and Baltimore, the call load initiated the collapse. The system was designed not to collapse; calls were designed to terminate correctly despite the originating call overload. But the terminating end also collapsed and aggravated the overload.

The system was designed to let the terminating call load progress; the software design required a relay to operate through a chain of 10 relays, which under overload were fibrillating. The relay would then allow calls to terminate properly. On paper, the design looked good, but the designer did not realize that the fast fibrillation rate precluded the relay from functioning. Overloads occurred rarely, and when they did, utter confusion prevailed.

Not until an extensive ECG-type analysis of the system was set up, and an overload was induced, was the "bug" found. The ECG type of tape showed that the relay received an op-

erating pulse but it was too short, similar to the effect on a heart that is set in fibrillation by an electric shock.

In those days of yore, glitches were known as "design bugs," so all we can say is that progress has simply changed the name to "software bugs."

*Maxwell G. Killoch
Retired AT&T executive
Fairhope, Alaska*

The moth in the relay of the Mark II, depicted on the cover of your Sept. 13 issue, is a hardware bug, not a software bug as the cover title might imply. Hardware bugs are still being produced, along with software bugs. There was an excellent one in an IBM PC clone I recently bought. The machine is back in the shop for the third time while the technicians try to find the crossed wires in the hard disk controller.

*Jackson W. Granholm
Thousand Oaks, Calif.*

Your cover bug reminded me of my own first system bug in the late fifties. A neon indicator refused to go out regardless of what was turned off or disconnected. The culprit turned out to be a mighty sick 100,000-ohm cockroach that bypassed all our logic and supplied current to the glow lamp directly from the incoming power line.

Having been involved with the computer industry for many years, I greatly appreciated the article on unsafe software and would like to pass on a few observations based on personal experience.

For example, there is the "sorcerer's apprentice" syndrome that can occur when a program operates so flawlessly that nobody watches it after a while. One such program was in charge of ordering material for producing relays and transformers from production forecasts and lead times. When the lead time for copper wire crept up from several months to a year and a half, the order rate was dutifully increased until the weight of the copper, stored off in a corner of the first floor of a 10-story building, caused the foundation to fail. (Fortunately it was an old, fully depreciated building, so another computer probably dismissed the event as fiscally insignificant.)

Ironically, the general acceptance of program bugs was pointed up by a program without bugs. Written to control a pick-and-place machine a few years ago, it worked perfectly the first time it was tested. The event was so noteworthy that a paper describing the program development was accepted and presented at a national CAD/CAM conference.

Lastly, a prediction: A major glitch will occur on the first business day after the turn of the century (00:01 Jan. 1, 2000). Many data bases and programs omit the "19" prefix from the year in their dates, and current entries will suddenly be 100 years older than previous inputs. Checks, being older than 90 days, will bounce. New mortgages, with no payments for nearly a century, will be foreclosed and cars repossessed, and new savings accounts will either receive an extra century's interest or be turned over to the state as dormant.

*Homer B. Clay
Phoenix, Ariz.*

Gardening with Wildflowers

By Frances Tenenbaum

A complete guide for beginners and experienced gardeners, including: General rules of wildflower culture, including the importance of mulch and compost; How to incorporate wildflowers into a traditional garden; How to use them in "problem" places —

along a driveway or in a back corner; How to create a small meadow or woods on a suburban plot; How to garden with native plants in a natural woodland, seashore or wetland site; Collecting plants from the wild to preserve them; Conservation rules and protected plants; How, where and when to buy wildflowers — plus a list of suppliers.



Ballantine, 1986,
165 pages, 6" x 9",
paperback, \$8.95

Science News Book Order Service
1719 N St., NW, Washington, DC 20036

Please send _____ copy(ies) of **Gardening with Wildflowers**. I include a check payable to Science News Book Order Service for \$8.95 plus \$1.00 handling (total \$9.95) for each copy. Domestic orders only.

Name _____

Address _____

City _____

State _____ Zip _____

RB603