

Predicting software bugs

Meir Lehman of Imperial College, London, wrote articles [see references] about software evolution. In one of them, he shows that the rate of bug-finding before release of the software product predicts the rate of bug-finding after release of the product. This is counter-intuitive, because we have a tendency to think that finding a lot of bugs during the QA and testing phase of a software project is good. After all, every bug found early in the project is one less bug to find later on.

*Unfortunately, according to Lehman, this is not a happy circumstance. **Finding a lot of bugs at any time during the project simply predicts that we will have a buggy product – forever.***

Lehman also observed that bugs are introduced into a software product as other bugs are fixed during maintenance of the product. The ratio of bugs introduced per bug fixed tends to grow with the size of the software. In the 1970s, Lehman estimated that software packages with more than one million lines of code could easily have a ratio greater than one, therefore the bug-fixing process diverged – the number of bugs in the product would increase without bound.

We can only hope that with modern object-oriented programming languages and better testing tools the number of lines of code at which the ratio exceeds one has gone up, because many current products have more than one million lines of code.

From Chapter 2 of
Get Out of the Way!
How to Manage Development
of Timely, Innovative, and Relevant Products
by John V. Levy, Ph.D.