

## Let's Update Software Reliability!

Norman Schneidewind  
Email: [ieeelife@yahoo.com](mailto:ieeelife@yahoo.com)

The field of software reliability needs to be more comprehensive in the types of systems it models. Up to this time, models have mainly addressed single node systems. While there have been notable improvements in prediction accuracy, the scope of the predictions has been myopic. Today, systems are less isolated and more and more dependent on the Internet to deliver services [1]. The implication of this change is that software reliability models must now address multiple nodes comprised of, for example, client, web server farms, search engines, and Internet routers. A major task in attempting to accomplish this is to find the Internet failure data to support the modeling effort. However, surprisingly, if one searches the Internet, it is possible to locate aggregate data on different types of Internet failures.

Another area software reliability should cover better is cyber security and the vulnerabilities and potential failures that are the results of vulnerabilities. A major source of vulnerability data is the National Vulnerability Database managed by the National Institute of Standards and Technology and sponsored by the Department of Homeland Security [2]. The database contains vulnerabilities by vendor, product, severity, type of access (client, local network, Internet), and date. If you make what I think is a reasonable assumption that the vulnerabilities lead to faults that lead to failures, you can compute approximate failure rates from these data.

A third area that could use some updating is to encompass the entire life cycle in modeling software reliability. Historically, the models have addressed the test phase or at most the test and operational stages, leaving out important phases like requirements analysis. For example, the ANSI/AIAA Recommended Practice of Software Reliability published in 1993 only addressed the test phase. Now, in the IEEE/AIAA update of this document, it is recognized that the seeds of reliability disasters are planted early in the life cycle by virtue of risky requirements [3].

### **References**

[1] Nevil Brownlee and kc claffy, "Internet Measurement", IEEE INTERNET COMPUTING [www.computer.org/internet/](http://www.computer.org/internet/) SEPTEMBER • OCTOBER 2004 33, pages 30-33.

[2] <http://nvd.nist.gov/statistics.cfm?vendorstart=2>.

[3] IEEE/AIAA P1633™/Draft 12b, Recommended Practice on Software Reliability, October 2007.