
Book Reviews

Book Review: **Fatal Defect: Chasing Killer Computer Bugs**

by [Saveen Reddy](#)

Ivars Peterson
1995
Random House, Inc. NY



As great a role as software plays in our lives, we should make an effort to be familiar with its positive and negative aspects. Increasingly, as computing becomes ubiquitous, many of us have had contact with the extreme consequences software failures can bring about. One of my own friends met his fiery end in Dhahran, Saudi Arabia when his barracks had the incredible misfortune of getting hit by a SCUD missile. My learning that a timing error in the Patriot's software had been partially responsible for his death underscored the kind of dependence we have on computers and the naive expectations we have about their correctness.

In *Fatal Defect* Ivars Peterson attempts to bring this issue to light through a careful examination of case studies and an exploration of what can be done to ameliorate the disastrous consequences of bugs in life-critical applications.

We have a tendency these days to exaggerate things. Everyday problems or rare injustices are interpreted as crises. Legitimate worries are cast as mass hysteria. With a book entitled *Fatal Defect* and chapters named ``Silent Death" and ``Time Bomb" I was originally apprehensive. I feared that the book would essentially be asking ``Computers: Threat or Menace?" I imagined that in the end it would recommend a Luddite retreat from the age of computing. I am pleased to have been wrong. He is fair in his treatment, not espousing complete trust in the abilities of people or machines.

A copious amount of time is spent detailing some disturbing case studies. However, for the most part Peterson does so calmly, rationally, and always emphasizing the human element. Rather than some dry, clinical description he manages to bring out a personal aspect to these stories. This is especially true in his coverage of the Therac-25 incidents, where patients were exposed to a highly energetic electron beam. The Therac-25 accounts are detailed and vivid, making it exactly clear what happened from the

victims' perspective.

Thankfully, there is more than just tragedy in these pages. In order to show why software is so difficult to control, Peterson deals with equally important issues such as software maintenance, numerical accuracy, reverse engineering, and software verification. The chapter on verification, entitled ``Absolute Proof," is perhaps the most interesting one (or at least the most humorous). A good portion of it deals with the Fetzer article on program verification in *Communications of the ACM* (September 1989) and the numerous, lengthy, and caustic responses it generated. Peterson allows us to bear witness to what amounts to the academic equivalent of a bar room brawl. The account is sure to be one of the most entertaining ones ever written on a subject that for the majority of normal people has all the excitement of watching paint dry.

The author also does the reader a great service by introducing, in some detail, the various players involved such as Peter G. Neumann, who is the moderator of the [RISKS forum](#), and Dave Parnas, known for his involvement with the SDI. Again, this brings out the human component of the story as opposed to technical minutiae. This approach makes for fine general reading, and that is his intention. Those who hunger for details may find it somewhat unfulfilling, though valuable as an overview of the subject. In any case, what it sacrifices in depth, it makes up for in its ability to hold the reader's interest.

Those who do want to explore the subject in greater detail should investigate Neumann's *Computer Related Risks*, ACM Press (Addison-Wesley).