

The Standish Report: Does It Really Describe a Software Crisis?

Reconsidering the relevancy of a frequently cited report on software project failures.

There has been plenty of discussion over the last several decades about something called “the software crisis.” Those who speak of such a crisis claim software projects are always over budget, behind schedule, and unreliable.

The software crisis thinking represents a damning condemnation of software practice. The picture it paints is of a field that cannot be relied upon to produce valid products.

But it is important to step back and ask some questions about this crisis thinking:

- Does it represent reality?
- Is it supported by research findings?

In this column, I want to make the point that, based on answers to these questions, there is something seriously flawed in software crisis thinking. The reality is, I would assert, that we are in the midst of what sociologists might

call the computing era—an era that would simply not be possible were it not for plentiful successful

software projects. Does that reality suggest the software field is really in crisis? Not according to my way of thinking.

Specifically, I want to address that second question, the one about research findings. At first glance, there are plenty of publi-

cations that conclude there really is such a crisis. Many academic studies assert the software crisis is the reason behind the concept the particular study is advocating, a concept that is intended to address and perhaps solve this purported crisis. Software gurus often engage in the same kind of advocacy, and also frame their pet topics as crisis solutions.

But there is an underlying problem here. Most such academic papers and guru reports cite the same source for their crisis concern—a study published by the Standish Group more than a decade ago, a study that reported huge failure rates, 70% or more, and minuscule success rates, a study that condemned software practice by the title they employed for the published version of their study, *The Chaos Report* [4].

So the Standish *Chaos Report* could be considered fundamental to most claims of crisis. What do we really know about that study?

That question is of increasing concern to the field. Several researchers, interested in pursuing the origins of this key data, have contacted Standish and asked for a description of their research process, a summary of their latest findings, and in general a scholarly discussion of the validity of the findings. They raise those issues because most research studies conducted by academic and industry researchers arrive at data largely inconsistent with the Standish findings.

Let me say that again. Objective research study findings do not, in general, support those Standish conclusions.

Repeatedly, those researchers who have queried Standish have been rebuffed in their quest. It is apparent that Standish has not intended, at least in the past, to share much of anything about where the data used for the *Chaos Report* comes from. And that, of course, brings the validity of those findings into question.

But now there is a significant new thought regarding those Standish findings. One pair of researchers [3], combing carefully over that original Standish report, found a key description of where those findings came from. The report says, in Standish's own words, "We then called and mailed a number of confidential surveys to a random sample of top IT executives, asking them to share failure stories."

Note the words at the end of

that sentence: "... share failure stories." If that was indeed the basis of the contact that Standish made with its survey participants, then the findings of the study are quite obviously biased toward reports of failure. And what does it mean if 70% of projects that are the subject of failure stories eventually failed? Not much.

There is a dramatic case of déjà vu here. In the 1980s it was popular to support the notion of a software crisis by citing the *GAO Study*, a report by the U.S. Government Accounting Office that described a terrible failure rate among studied software projects. But in that case, after this had been going on for far too long, one alert researcher [1] reread the *GAO Study* and found that it admitted, quite openly, that it was a study of projects known to be failing at the time the data was gathered. Once this problem was identified, the *GAO Study* was quite quickly dropped as a citation to support the notion of software crisis. It is interesting that the first Standish study came along not too long afterward.

Is it true that the Standish study findings are as biased toward failure as the *GAO Study* results? The truth of the matter is, we don't really know. That quoted sentence cited previously certainly suggests so, but it is not at all clear how much of the study was based on the initial contact that sentence describes. And how

much of the subsequent study findings (Standish has repeated its survey and updated its *Chaos Report* several times over the ensuing years, see [2]) were also based on that same research approach?

Once again, it is important to note that all attempts to contact Standish about this issue, to get to the heart of this critical matter, have been unsuccessful. Here, in this column, I would like to renew that line of inquiry. Standish, please tell us whether the data we have all been quoting for more than a decade really means what some have been saying it means. It is too important a topic to have such a high degree of uncertainty associated with it. **C**

REFERENCES

1. Blum, B.I. Some very famous statistics. *The Software Practitioner* (Mar. 1991).
2. Glass, R.L. IT failure rates—70 percent or 10–15 percent? *IEEE Software* 22, 3 (May–June 2005).
3. Jorgensen, M. and Molokken, K. How large are software cost overruns? A review of the 1994 *Chaos Report*. *Information and Software Technology* 48, 4 (Apr. 2006).
4. Standish Group International. *The Chaos Report*; www.standishgroup.com/sample_research/PDFpages/Chaos1994.pdf.

ROBERT L. GLASS (rlglass@acm.org) is the publisher/editor of *The Software Practitioner* newsletter and editor emeritus of Elsevier's *Journal of Systems and Software*. He is currently an honorary professor in the ARC Center for Complex Systems at Griffith University, Brisbane, Australia.

Copyright of Communications of the ACM is the property of Association for Computing Machinery and its content may not be copied or emailed to multiple sites or posted to a listserv without the copyright holder's express written permission. However, users may print, download, or email articles for individual use.