

A Government-Controlled United States Software/IT Industry?

Jeffrey Voas

Reliable Software Technologies, Sterling VA USA

1 Introduction

Sound absurd? Well it shouldn't. If the US Government decides to break Microsoft into smaller companies, such action will demonstrate the US Government's willingness to legislate our industry when it feels that the industry (or a company within the industry) wields too much control.

Let's also consider the impact that software and IT has on world economics. The technologies our industry provides control so many aspects of everyday life in numerous countries around the world. Intuition suggests that governments may someday feel the need to regulate the IT industry simply because of the power that IT has on those governments. After all, CIOs and CEOs of the IT industry aren't supposed to be in charge: the politicians are.

In this column, I'd like to focus on a different reason why government control of the IT industry may occur: (1) public fear of software and computers, and (2) our well-publicized software quality problems throughout the industry. In my opinion, these issues are sufficient to turn us into a regulated industry.

To begin, society generally distrusts information systems. Few people understand how computers work, and that creates a myth that our industry produces products that invade personal privacy and take away personal control. Also, we are experiencing a public relations nightmare as a result of the Year 2000 problem. Our industry is viewed as the party responsible for creating this mess, and many people think that the Year 2000 problem will lead to global crisis.

Further, software is a key ingredient in nations' infrastructures, including national security, banking, and communications. When such software fails or is successfully attacked (because of poor security), governments become vulnerable. Do you think governments will allow themselves to be held hostage to defective or insecure information technology?

Recognize that government regulation over software quality is not a *revolutionary* idea. The US government already regulates software quality in certain industries. For example, Boeing and Airbus do not have the latitude to develop and test the software on their aircraft using whatever processes and standards they prefer. Mandated rules (in the standard known as DO-178B) are required by the Federal Aviation Administration concerning how software on commercial aircraft is developed and tested. It is clear why governments regulate software in safety-critical applications such as avionics; people must trust aircraft and be willing to fly on

them. But what about non-safety-critical applications? If the government decides to regulate the quality of these systems, that will represent an *evolutionary* change in policy.

2 What Do The Experts Say?

I fear that the US Government will, at some point, decide to regulate non-safety-critical information systems. Why? Because non-safety-critical systems control numerous aspects of everyday life, and although those systems do not directly threaten lives, it is nonetheless vital that those systems operate properly. But that is just my opinion. Let's now consider what the experts are saying on this matter.

The First International Software Assurance Certification Conference (ISACC'99) was held in the Washington D.C. area on March 1-2, 1999. The conference focused on different philosophies on how to certify that software is reliable, safe, secure, etc. Most speakers discussed the main three approaches: (1) certifying software products, (2) accrediting personnel, and (3) auditing a publisher's process maturity [1].

In this column, I'd like to replay what three software/IT luminaries stated at ISACC'99 on the issue of whether the government will someday try to regulate software quality certification laws: (1) Gary Beech (Publisher of CIO magazine), (2) Harris Miller (CEO of the Information Technology Association of America), and (3) Ed Yourdoun (CEO of the Cutter Consortium). Interestingly, all admitted that increased government control is a real possibility. While their reasoning behind what it would take to make this happen differed, their agreement on this matter was noteworthy.

Beech stated that the outcome of the Microsoft *vs.* US Government anti-trust case will be pivotal in determining whether the government begins taking control. During his talk, Beech presented an interesting insight on how current levels of software quality were affecting CIOs around the country. He stated that the ubiquitous lack of software quality was definitely being felt by "his end-users" (America's CIOs). He also went on to state that CIOs were so tired of software quality problems that he felt they would be willing to pay more for software if they got more quality. And if CIOs are feeling the effects of poor quality, so are government organizations. Toward the end of his keynote speech, Beech called for an industry-led initiative to address how software should be certified.

Harris Miller seemed the most concerned about the possibility of government intervention. He discussed a hypothetical scenario where software companies might someday be forced to have their software tested in government labs before licensing it commercially. He pointed out that such regulation could add an extra nine months onto the date for product releases. He urged the audience and the industry to get the jump on the government by being the first to create adequate certification and testing schemes.

Miller's rationale stemmed from IT being around 6% of the global GNP and 8% of the United State's GNP. He argued that at this size, the IT industry now has the political clout that industries such as the automotive industry has held for years. But he warned that with these rights come responsibilities, and with global economies based so heavily on the IT industry (not only because of the size of our revenues because of what our products and services do for the remaining industries), politicians will not risk allowing the IT industry to

make quality-related blunders that could seriously hurt economies, citizens, and nations.

Ed Yourdon voiced a similar position. Yourdon strongly argued that government certification *will occur, but only after* a widespread calamity occurs that is directly attributal to software failure. Yourdon went on to state that if the Year 2000 software problem winds up being that problem, people are likely to blame politicians and governments as much if not more than the IT industry. If true, politicians will have to champion platforms calling for the regulation of the IT industry.

Yourdon stated that such platforms and rhetoric could be heard as early as the Year 2000 Presidential race. As most of you know, Yourdon is highly concerned about the potential of serious social problems occurring as a result of improperly remediated code. If Yourdon is right on both issues, we could see the government beginning to create certification/regulation schemes as early as 2000.

I agree that as long as no serious software-related disasters occur, the government is likely to keep its distance from the industry. After all, planes are not falling out of the sky, and banks are not failing due to software failures. Politicians in the United States are far more intrigued with issues such as eliminating the marriage-penalty tax, providing a middle-class tax cut, and shoring up the social-security trust fund. In short, IT legislation is not a hot political issue because “Joe Citizen” is not demanding it.

3 Summary

I am unaware of another profession that disclaims liability in the manner that software industry does. While I believe that government regulation and government certification of software quality is unlikely to make non-safety-critical software substantially better (and is only likely to raise the cost of software and slowdown new releases), such actions provide platforms for politicians who can then claim that they are taming an uncaring industry.

How soon this might happen is also unclear. If the IT industry meets its ethical obligations to produce quality services and products, it is hard to imagine why the government would step in. But as we all know, quality is not something most people think of when they think about the IT industry. Most people think of rebooting computers, loosing files, viruses, invasion of privacy through the Internet, e-mail being lost, networks being down, etc. Because these problems have not led to disasters, our industry has have gotten away with it.

If there is a widespread software-related disaster, however, the public will demand government action. And ultimately it will have been our fault, because we are the ones that pushed software into every aspect of peoples lives and then invented licensing schemes that disclaimed responsibility when the software failed.

References

- [1] J. VOAS. The Software Quality Certification Triangle. *Crosstalk*, 11(11):12–14, November 1998.