



# LITIGATION VS. INNOVATION

Let's strike a balance between protecting intellectual rights and encouraging innovation

The personal computer revolution was catalyzed by young rebels, dropouts, and visionaries who helped forge what has become a multibillion-dollar industry of global import. These people fundamentally changed how computers were used and how business—and government—got work done, by making computers useful and accessible to millions of ordinary citizens without special computer expertise. I like to think that the company I founded, Lotus Development, played an important role in that.

As a software entrepreneur, my perspective on intellectual property isn't grounded in theory or law—it's based on my experiences trying to turn innovative ideas into real businesses.

Let me make a bias clear up front: I like new ideas. I like being part of a creative community of software designers, each trying to surpass the others. I like having the market tell us which innovations are exciting and which aren't. I like the fact that copying and distributing pirate versions of our software is a punishable crime.

Let me tell you what I don't like. I don't like companies acting as if they have a monopoly on a good idea. I don't like companies forgetting that, like it or not, they also learn from their competitors and their competitors' customers.

Pamela Samuelson, an intellectual-property scholar at Emory University in Atlanta, describes intellectual-property advocates as ranging along a scale from minimalists, who believe in the bare essentials of protection, to maximalists, who insist that intellectual property is so

precious that it must be surrounded by a phalanx of razor-edged laws.

I am a minimalist. That doesn't mean I don't care about intellectual-property protection; it means that I don't want protection to become the dominant theme, or even a dominant theme, of this industry. If you want to keep this industry as vibrant and successful as it's been, then a properly constructed intellectual-property policy will respect protection but give preference to innovation. Overprotection of intellectual property is as pernicious as underprotection in its stifling effects on innovation and the consequent loss to society.

Unfortunately, the computer industry is experiencing an unsteady but stubborn march to extend the scope of copyright. Twisting and straining each step of the way to secure additional copyright protections, too many companies seem to have decided that it's easier to sue their rivals than compete with them. Litigation is becoming a business tactic, not a practice of last resort. Software should not be an industry driven by litigation. That would be bad for both the industry and its millions of customers.

It would be great if we could just draw a line and, say, outlaw software clones of specific application programs. But I'm concerned about where the line ultimately gets drawn. The next foreseeable step, in which litigants seek to protect individual features and elements of programs, per se, under copyright, would be one step too far.

Speaking from my own observation, the so-called spreadsheet clones have achieved but the tiniest of market shares, and I don't believe that's an accident. Cloning applications is an unviable business strategy. Success in the software business depends on many factors: documentation, training, customer support, and the quality of customer relations in general. All these factors favor the large, well-financed software companies.

Software is complex and idiosyncratic;

unless someone is deliberately copying the internals of the code, reproducing a sophisticated application with quality and utility equivalent to the original is difficult and expensive. Any firm with the resources to do a good job at this prefers to create original products that represent a greater opportunity.

It's the nature of software for ideas to slosh and flow back and forth between competitors, companies, and industries. Like architecture and the movies, software is a medium for ideas.

Some firms would like to have their works fully protected but be free to benefit from the efforts of others without much regard for intellectual-property rights. They must be reminded that the law has an obligation to be evenhanded.

Of course, complicating all this is that software is a different kind of intellectual-property beast. Professor Samuelson observes that software is both a writing and a machine—in a legal system that has assumed something can be either a writing or a machine, but not both.

Increasingly, the economic value that we add to this society and the global economy is this intangible, crystallized mindstuff called software. America's software industry happens to be the best in the world, and that isn't due to intellectual-property lawsuits. The challenge is, what regime is going to continue to support our ability to do well? If our policy comes out of court battles, then we're going to have an industry that looks as though it were shaped by lawyers and judges, not by technically innovative and market-sensitive entrepreneurs. ■

*Mitch Kapor, founder of Lotus Development Corp. and co-creator of Lotus 1-2-3, is president of On Technology. This column is adapted from testimony offered to the congressional committee on intellectual-property rights, to which he made the opening remark, "Software has been very, very good to me." He can be reached on BIX c/o "editors."*