

# Lotus Disinformation

## Forewarned is Forearmed

(Revised and extended, 7 February 1991)

Many of the statements by Lotus on the issue of interface copyright are designed to mislead you in specific ways. There are two basic facts that Lotus does not want the public to understand:

- That user interface copyright is a recent change in the legal system.
- That imitating the commands of a program is different from copying the program itself, and that until recently these were regarded differently both by the legal system and by nearly all developers of software.

To achieve this aim, Lotus spokespersons refuse to recognize the distinction between copyright on user interfaces and copyright on programs; they speak only of “copyright” without differentiation. They use the term “copying” to describe both copying disks by machine and implementing new, compatible programs. This may lead unsuspecting readers to regard these as equivalent, overlooking the difference.

You can avoid propagating confusion by being alert and recognizing these implicit messages when you interview Lotus representatives—and explaining the flaws in the statements that you quote.

Past misleading statements made by Lotus include:

- *“It [copyright] outlaws unfair competition by those who copy and sell the creations of others.”*

The Lotus lawsuits are not about copying programs, but about imitating features that are popular with users. Imitation was generally accepted custom—something that Lotus itself practiced—until the first surprise “look and feel” lawsuits began a few years ago. It was considered both fair and lawful.

- *“Copyright has been fundamental to the development of the computer industry.”*

This may be true of the kind of copyright that existed during the development of the industry, which was copyright on individual programs. (The League for Programming Freedom does not oppose this traditional interpretation of copyright law.) It cannot be true of copyright on user commands, which is a recent change.

- *“Judge Keeton’s decision did not change the law.”*

Not by itself; but it is the last of a series of “look and feel” decisions which have done so. The first of these decisions, a few years ago, surprised the entire computer industry. It is not meaningful to ask which single decision of the series was the one which changed the law.

- *Opponents of interface copyright are academics, “ignorant of the realities of the marketplace.”*

The members of the League for Programming Freedom include successful entrepreneurs, executives and independent consultants, as well as numerous programmers employed in industry.

- *“The argument for weak protection is coming essentially from two places. It is coming from large Japanese hardware companies... The other supporters of weak protection typically are some of the third world countries.”*

Sun Microsystems, a \$2 billion US company, advocates a narrow interpretation of copyright. So do the members of the European Committee for Interoperable Systems (ECIS), a group of over sixty companies, user groups, and industry associations. Its members include Bull S.A., Ing. Olivetti S.A., N.C.R. Corporation, Fujitsu, Unisys, Sun, and McDonnell Douglas Information Systems, and others.

- *Comparison of copyright on user interfaces with copyright on books and movies.*

Learning the interface of a program represents a substantial investment by each user. Few users are willing to make this investment a second time to learn to use an alternative program, even a superior one. This presents a barrier to entry for competitors that has no equivalent in the world of books and movies.

Readers do not reject a new novel because it is different from the one book they have already learned to read—this would be absurd. But computer users do reject new user interfaces for precisely that reason.

Only a few aficionados learn program interfaces for their own sake. Most users learn them in order to use a program. By contrast, we read a book in order to see what it says; there’s nothing more to it. The contents of a book are analogous to particular information that a program or database might tell the user. They are not analogous to the language in which data is conveyed to or from the user.

There is one aspect of a book which is analogous to the user interface of a program. This aspect consists of the language it is written in, plus the conventions of book layout (such as chapters, page numbers, indices, and tables of contents). These things are what a user must know in order to use the book. They are not copyrightable.

- *“No one has forced anyone to buy 1-2-3.”*

People are often forced by circumstances to use a spreadsheet with the commands of 1-2-3. For example, many large corporations adopt internal standards, to reduce problems of compatibility and training. Beyond this, the fact that the interface of 1-2-3 is a standard means that other alternatives are of no practical use for most users.

These are the same factors which effectively compel nearly all typists to use the de facto standard QWERTY keyboard. To a person compelled in this way, quibbling about the meaning of “force” is little consolation.

- *“So far we have sued only those who have copied the entire interface.”*

Users who have invested time in learning the interface of 1-2-3 are entitled to the option of competing programs which do not require retraining. This means, programs that can understand all the commands of 1-2-3.

Meanwhile, Lotus refuses to rule out suits in the future against programs that are only partly compatible. Copyright applies to works that are only partly similar; without a

change in the law, interface copyright will prohibit imitating any substantial part of an interface.

- *User interface copyright will promote innovation.*

These copyrights will not promote meaningful innovations. Rather, they will promote pointless change comparable to scrambling the keys on the typewriter keyboard. They will stifle true innovation based on incremental improvements, because lawsuits against creators of a program with any significant similarity to an earlier program will be an ever-present threat.

- *We should let the judicial system decide the issue, and not interfere.*

Just recently, a bill was passed to prohibit rental of software; software publishers including Lotus strongly supported this bill. Lotus tries, as do the rest of us, to change laws that seem incorrect. They say that pushing for change in the law is wrong, hypocritically, when they want the laws to stay the same.

The judicial system is subject to “garbage in, garbage out” like any other system. If we find that the courts handle certain issues badly, we should not hesitate to ask our elected representatives to change the laws that judges follow. This is the meaning of government of the people, by the people and for the people.

- *“New laws take a lot of time to draft, and no matter how well we draft the new law around software protection, it’s going to take 20 or 30 years of case law in that domain to make it work.”*

The implication is that case law has already progressed five years of that required time, and we shouldn’t sacrifice that by starting over with a new law.

The fallacy here is the assumption that case law inevitably reaches a proper solution. In fact, judges believe it is their job to follow the orders given in the existing laws—but in a rather mechanical fashion. When this directs judges in the wrong direction, when the law as interpreted is fundamentally wrong, we cannot expect further judicial clarification to make it right. This law simply needs to be replaced.

- *“I don’t want somebody ripping off something I worked hard to produce.”*

The user interface of most programs, including 1-2-3, was far less work to develop than the program itself. The main reason the interface of 1-2-3 is valuable is that so many users have invested time and money in it. So much for “hard work”.

- *“There is no innovation in copying software.”*

“Copying software” means copying a program from one disk to another, and is deceptive when used to describe the act of implementing a compatible program.

However, this statement makes an additional false implication: that developing a program with compatible features and commands involves no innovation.

In fact, the user interface is just the surface of a program; the entire internal workings of a compatible imitation program must be developed from scratch. The new program may be an improvement in speed, reliability, resources needed, makes and models of computer supported, terms of availability, or price.

Typewriter manufacturers such as IBM and Olivetti have made typewriters that could be considered “clones” of earlier models, since they use the same keyboard layout. However, it is not generally felt that designing their products involved no innovation, or that they should have been prohibited.

Even to speak of “innovation” is misleading, because it sets up a double standard. We do not require innovation in books or movies. For example, most gothic romance novels contain little real innovation, but that does not mean they constitute plagiarism.

- “... *protection of innovation from being ripped off by others [is guaranteed] ... under the Constitution.*”

This statement appeals to a common misconception about the purpose of copyright law. The purpose, according to the Constitution and the Supreme Court, is not to guarantee special entitlements to authors. Rather, it is to promote progress in science and the arts, for the sake of the public.

Granting an overlarge monopoly to anyone works against progress and the free enterprise system. Much as Lotus may be distressed when their work contributes to wider progress, there is no public interest in preventing this.

- “*As years go by, we will find that interface ... will be harder and harder to separate from the content and value of software.*”

The difference between a program’s implementation and its interface is not a technological one. It is the difference between how the program works and how users tell it what to do. This distinction is a matter of common sense; technological changes cannot erase it. A similarity limited to how users tell a program what to do should not be covered by copyright.