UIC John Marshall Journal of Information Technology & Privacy Law

Volume 14
Issue 4 *Journal of Computer & Information Law - Summer 1996*

Article 1

Summer 1996

Intellectual Property Protection of Software: International Perspectives: Foreword, 14 J. Marshall J. Computer & Info. L. 657 (1996)

Donald P. Reynolds

INTELLECTUAL PROPERTY PROTECTION OF SOFTWARE: INTERNATIONAL PERSPECTIVES

FOREWORD

by DONALD P. REYNOLDS†

Ten years ago the predecessor to the John Marshall Journal of Computer and Information Law published a symposium issue on the legal protection of computer software over the world. That predecessor was the Software Law Journal, and its issue for the Spring-Summer of 1986 was a symposium that was an ambitious undertaking. The issue reported on the protection of computer software in fourteen countries placed in alphabetical order. This, of course, put the United States in last position, an appropriate one for an excellent summary by my colleague Professor Raymond Nimmer of the University of Houston Law Center. The symposium issue of 1986 also had articles on the patentability of software-related inventions under the European Patent Convention, copyright protection of computer software, and the legal protection of semiconductor chips in the United Kingdom. All in all, the symposium issue was a significant undertaking. It is interesting to compare the software world today with the world that existed then.

Personal computers existed in 1986, but they were not the widely-available tools that they are today. For the most part, software worth protecting in 1986 was written for main-frame computers, and its protection was appropriate to the existence of a relatively small number of potential infringers, whether of copyrights or patents. Infringers who

† Donald P. Reynolds is the Director of the Center for Intellectual Property and Professor of Law at The John Marshall Law School. Professor Reynolds is a member of Tau Beta Pi, Eta Kappa Nu, Sigma Xi, and Phi Delta Phi. He is admitted to practice in New York and Illinois and is registered to practice before the U.S. Patent and Trademark Office. He is active in the Intellectual Property Law Association of Chicago and the American Intellectual Property Law Association, as well as other professional associations. He is a senior member of the Institute of Electrical and Electronic Engineers and is a member of the Executive Committee of its Chicago Section. He obtained his B.E.E, M.E.E and his J.D. from Cornell University. He is of counsel to the Chicago firm of McAndrews, Held & Malloy.

operated main-frame computers were generally solvent and stationary. If you could catch them as copiers of your software, you could expect to recover substantial amounts of money in damages. Either this fact or a generally adequate standard of business ethics led to a relatively small number of acts of infringement.

Today many people have personal computers that approach or exceed the capabilities of main-frame computers of 1986. This fact, and the explosive development of the Internet, have created problems that are just beginning to be understood, let alone solved. It has never been so easy to steal copyrighted material, including software, and to publish it on the Internet for the world to use, often without any knowledge that the software is stolen. A selective understanding of copyright law and the First Amendment to the United States Constitution also causes some people to find virtue in such stealing, and many of these people are undetectable and judgment-proof.

Perhaps an indication of the pervasiveness of the legal questions relating to software can be determined by noting the 1996 case *Lotus Development Corp. v. Borland International*,[1] in which the United States Supreme Court divided four to four to allow a decision of the First Circuit to stand. In that decision, a set of instructions totalling over four hundred in number was held not to be copyrightable subject matter because the instructions merely told the computer how to carry out its functions. This case adds the question of appropriate subject matter for copyright, one software developers had not worried much about, to the question of how do I protect the software I have written from being copied or paraphrased. Software developers and other producers of copyrightable material often disagree with the courts which hold that the copyright protects its owner from literal copying of a writing but does not protect the ideas contained in the writing. This is somewhat easier to accept if you are the writer of a pamphlet than it is if you write software, in which it often seems that the expression is the idea. If this is the case, appropriating the idea appropriates the expression.

The United States Supreme Court has also created an additional problem in the area of copyrightable subject matter in *Feist Publications v. Rural Telephone Service*,[2] a case not involving software, in which the Court stated expressly that the "sweat of the brow" did not render a compilation copyrightable subject matter. This represents a potential threat to those who generate software in that they may find their creativity challenged after the fact. So far, though, software has generally been regarded as copyrightable subject matter in the United States, and the rest of the world has agreed. The problem is instead how do you enforce

1. Lotus Development Corp. v. Borland Int'l, Inc., 116 U.S. 804 (1996).
2. Feist Publications, Inc. v. Rural Telephone Service Co., Inc., 499 U.S. 340 (1991).

your copyright on software? How do you detect infringement; how do you stop it when you find it; and how do you get damages from the people who put your software on the Internet and those who use it after they get it from the Internet?

Looking back again to 1986, those of us who were not computer hobbyists were not aware of computer hackers then. We did not have bulletin boards. Our modems, if we had them, were expensive and limited in capacity, and viruses were things that made us sick. There was no World Wide Web, no clearance of domain names, and, for all but a very few, no uploading or downloading to and from personal computers.

The present issue discusses some software problems of today in the United States and a number of foreign countries. Patent protection of computer programs — actually, the protection of patentable inventions embodied wholly or partly in computer programs — is generally accepted throughout the world, with some variation in the standards of invention from country. Copyright continues to be the principal source of protection for computer programs, with a 1993 law returning Germany to the level of protection of the rest of the world. Belgium has enacted a Software Protection Act that is more specifically directed to the protection of software than the laws of most other countries, which extend their general copyright laws to software as copyrightable subject matter. The People's Republic of China has in recent years joined the rest of the world in its legislative approach to protection of software by copyright, although the lack of a national enforcement mechanism still leads to concerns in other nations about the effective level of protection of their software. Most enforcement in China is in provincial courts which vary in effectiveness.

Trade secret protection is available in most of these countries where there is a possibility of making the software available and still keeping it a secret. This protection too is easy for the thief to subvert by use of the Internet to disseminate stolen software. As with copyright protection, it is easier to define rights than to enforce them against the pirate who is often difficult to catch and who often lacks the assets to answer to the liability resulting from being caught stealing.

We live in an exciting time. It may be more than exciting for computer hardware manufacturers, who see prices driven continually downward by fierce competition. The result for users is an amazing increase in the availability of computing power, memory, modem baud rates, and the software and graphics to use them, all at prices that are increasingly affordable. We need to be able to protect the creators of the software that makes these results possible, and in particular we need to be able to establish some measure of responsibility for use of the Internet without sacrificing the freedom it gives us to communicate as we never could before. The future will bring us more, and it is as hard today to feel

assurance in predicting how it will do so. Think about ten years ago and consider how unlikely today's computer world would have seemed. I will predict only that things will get better in ways we have not yet considered.