Marquette Intellectual Property Law Review

Volume 2 | Issue 1 Article 5

1998

No Protection, No Progress for Graphical User Interfaces

Jane M. Rolling

No Protection, No Progress For Graphical User Interfaces

INTRODUCTION

Innovation in the technical industries is fueled by competition.[1] As with any other industry, the key to successfully competing in software is capable research and development efforts; the end result of which is evidenced by the ability to roll out innovative and effective software products. The incentive for software producers, of course, is the ability to recoup the costs of research and development and to realize profits. Integral to this process of applying significant resources and recouping profits is the ability to use intellectual property laws to protect innovative products from competitors. Without such protection, the economic incentive to innovate and develop is gone.

There has, however, been a general reluctance by courts to grant any kind of intellectual property protection for visual elements of computer software, called graphical user interfaces.[2] Consequently, there have been no real developments since 1984, when the first graphical software program was introduced. The failure to offer protection for

---

1. *See, e.g.,* Roger B. Andewelt, *Antitrust Perspective on Intellectual Property Protection*, Remark to the American Bar Association (July 16, 1985), *reprinted in* 30 PAT. TRADEMARK & COPYRIGHT J. (BNA) 319 (1985). Justice Department official Roger Andewelt commented that intellectual property protection can promote competition stating, "[intellectual property] protection can encourage firms to compete through the development of new technologies and thereby can result in additional choices for consumers; new and better products and services and cheaper prices."

2. *See, e.g.*, Sega Enters., Ltd. v. Accolade, Inc., 977 F.2d 1510, 24 U.S.P.Q.2d (BNA) 1561 (9th Cir. 1992) (court did not want to allow a trademark owner to enjoy a monopoly over the functional use of the mark); Computer Assocs. Int'l v. Altai, Inc., 982 F.2d 693 (2d Cir. 1992) (holding that nonliteral portion of a computer software program is not copyrightable); Interactive Network v. NTN Communications, 875 F. Supp. 1398, 34 U.S.P.Q.2d (BNA) 1380 (N.D. Cal. 1995) (finding no trade dress protection for user interface because it is functional); Apple Computer, Inc. v. Microsoft Corp., 799 F. Supp. 1006, 24 U.S.P.Q.2d (BNA) 1081 (N.D. Cal. 1992) (holding that Apple's display elements were functional and therefore not protectible under copyright law); Digital Communications Assocs., Inc. v. Softklone Distrib. Corp., 659 F. Supp. 449, 2 U.S.P.Q.2d (BNA) 1385 (N.D. Ga. 1987) (computer programs are copyrightable as literary works but copyright protection does not extend to the screen displays). *But cf.* Lotus Dev. Corp. v. Paperback Software Int'l, 740 F. Supp. 37, 15 U.S.P.Q.2d (BNA) 1577 (D. Mass. 1990) (holding that Lotus is entitled to protect the format including the menus and screen of the software program); *see generally* Mark A. Lemley, *Convergence in the Law of Software Copyright?*, 10 HIGH TECH. L.J. 1 (1995) (setting forth a history of copyright protection).

computer user interface elements, and the resulting lack of development in this area, has been counterproductive to consumers and to the software industry.[3] Until a consistent and reliable method of protecting graphical user interfaces is applied by courts[4] (to spark competition and in turn, to promote technological progress,)[5] computer users, for the most part, are relegated to the uniformity that we see now, and have seen for the last fifteen years.[6]

Respected commentators from both legal and technical fields have spoken out in favor of a new paradigm for the protection of computer software.[7] The call for a new system of laws is premised on the assertion that computer software is so unique and the problems so distinctive that

---

3. Respected commentators in the field have gone so far as to state that competitors and society benefit more from a firm's innovation than the firm itself does. *See generally* Anthony L. Clapes, *Software, Copyright and Competition: The Use of Antitrust Theory to Undercut Copyright Protection for Computer Programs,* (414 PLI Patents, Copyrights, Trademarks, and Literary Prop. Course Handbook Series No. 553 (1995)); E. Mansfield, *How Rapidly Does New Industrial Technology Leak Out?*, 34 J. INDUS. ECON. 217 (1985).

4. *See, e.g.*, Apple Computer v. Microsoft Corp., 35 F.3d 1435, 32 U.S.P.Q.2d (BNA) 1086 (9th Cir. 1994); Engineering Dynamics, Inc. v. Structural Software, Inc., 26 F.3d 1335, 31 U.S.P.Q.2d (BNA) 1641 (5th Cir. 1994); Apple Computer, Inc. v. Microsoft Corp., 821 F. Supp. 616, 27 U.S.P.Q.2d (BNA) 1081 (N.D. Cal. 1993); Apple Computer, Inc. v. Microsoft Corp., 759 F. Supp. 1444, 18 U.S.P.Q.2d (BNA) 1097 (N.D. Cal. 1991); Lotus Dev. Corp. v. Paperback Software Int'l, 740 F. Supp. 37, 15 U.S.P.Q.2d (BNA) 1577 (D. Mass. 1990); Digital Communications Assocs., Inc. v. Softklone Distrib. Corp., 659 F. Supp. 449, 2 U.S.P.Q.2d (BNA) 1385 (N.D. GA 1987); Broderbund Software, Inc. v. Unison World, Inc., 648 F. Supp. 1127, 231 U.S.P.Q. (BNA) 700 (N.D. Cal. 1986).

5. Apple Computer, Inc. v. Microsoft Corp., 35 F.3d 1435, 32 U.S.P.Q.2d (BNA) 1086 (9th Cir. 1994); Xerox Corp. v. Apple Computer, Inc., 34 F. Supp 1542, 14 U.S.P.Q.2d (BNA) 1512 (N.D. Cal. 1990); *see also* Bill Curtis, *Engineering Computer "Look and Feel": User Interface Technology And Human Factors Engineering,* 30 JURIMETRICS J. 51, 76 (1989) ("The least excusable reason for interface similarities is ordinary laziness.").

6. The first widely marketed personal computer with software using a sophisticated, but simple to operate, graphical user interface was the Macintosh, introduced by Apple Corp. on January 24, 1984. *Triumph of the Nerds, Parts I, II, & III,* (PBS television broadcast, August 26, 1997). Transcript of the program also available at <http://www.pbs.org/nerds/transcript.html> [hereinafter Transcript]. *Id.*, at Part III.

7. *See* Pamela Samuelson et al., *A Manifesto Concerning he Legal Protection of Computer Programs*, 94 COLUM. L. REV. 2308, 2365-2420, 2429-31 (1994)[hereinafter *Manifesto*] (both legal and technical authors contributed to this article); Ken Liebman & Gary Frischling, *The Shape of Things to Come: Design-Patent Protection For Computers,* 9 No. 11 COMPUTER LAW. 1 (1992); Irwin R. Gross, *A New Framework For Software Protection: Distinguishing Between Interactive And Non-Interactive Aspects Of Computer Programs,* 20 RUTGERS COMPUTER & TECH. L.J. 107 (1994); Mark Aaron Paley, *A Model Software Petite Patent Act,* 12 SANTA CLARA COMPUTER & HIGH TECH. L.J. 301 (1996); Pamela Samuelson, *Counterpoint: An Entirely New Legal Regime Is Needed,* 12 No. 2 COMPUTER LAW. 11 (1995).

only a specific application and reconstruction of law will do.[8] There are inherent problems with creating a third paradigm for the protection of software, however,[9] and the creation of such becomes largely unnecessary when one considers promulgated and judicially developed laws that are applicable to user interface elements.[10]

Trade dress law, a judicially created subset of trademark law, is a theoretically and practically sound manner of protecting graphical user interface elements. This Comment supports the proposition that graphical user interfaces, taken as a whole, indicate the source of the product and are therefore deserving of protection under trade dress principles. In Part I, the historical context of graphical user interfaces and traditional methods of legal protection are described. Part II provides a summary of the unique qualities of computer software that make legal protection challenging. Part III sets forth the premise that current methods of protecting graphical user interfaces are inadequate, and the conclusion drawn that trade dress law provides the protection that software user interfaces demand.

I. SHORT HISTORY OF SOFTWARE AND HOW THE LAW HAS ATTEMPTED TO PROTECT GRAPHICAL USER INTERFACES

A. *Paper Tape to Complex Graphics*

"Even the best technology won't be used if it is too complicated."[11]

---

8. *See, e.g., Manifesto, supra* note 7, at 2378-2405.

9. *See* Jane C. Ginsburg, *Four Reasons and a Paradox: The Manifest Superiority of Copyright Over Sui Generis Protection Of Computer Software*, 94 COLUM. L. REV. 2559, 2559-60 (1994) (the computer industry is thriving despite the *Manifesto* authors' charge that current copyright protection is inadequate).

10. *See generally* Lauren Fisher Kellner, *Computer User Interfaces: Trade Dress Protection for Look and Feel*, 84 TRADEMARK REP. 337 (1994); H. Dale Langley Jr., *Tamale Sales and Carburetor Tune-Ups: The Implication for Computer Software Developers and Owners—A Trade Dress Analysis*, 3 TEX. INTELL. PROP. L.J. 9 (1994); Rhoda L. Rudnick, *Window Dressing: Trademark Protection for Computer Screen Displays and Software*, 80 TRADEMARK REP. 382 (1990); Willajeanne F. Mclean, *Opening Another Can of Worms: Protecting Product Configuration As Trade Dress*, 66 U. CIN. L. REV. 119 (1997).

11. Rebecca Quick, *Just Like Use: To Be Truly Useful, Computers Are Going To Have To Start Acting A Lot More Human*, WALL ST. J., June 16, 1997, at R14; *see also* Walter S. Mossberg, *Two PC Programs Offer Inexpensive Help with Tax Headaches*, WALL ST. J., March 5, 1998, at B1 ("[I] regard today's personal computers and software products as unacceptably complex. The instructions on how to use them are generally impenetrable, and the organizations who design them are too unresponsive to their users.").

At first glance, the concept of "user friendliness" sounds trite. In reality, however, the manner in which a computer interacts with its operator largely dictates the commercial success of the software program, or lack thereof.[12] The outward behavior of the software, called the graphical user interface, is the concept behind a computer program's ability to interact with a user in a way that is productive, intuitive, and efficient.[13]

Computers and their users interact through the use of "software," which was formerly a tape made out of paper and contained strategically placed punched holes.[14] The computer user was required to perform a series of complex numerical calculations in order to effect a mathematical code that represented the problem, and the code was then transferred in the form of holes onto the paper tape which provided instruction to the computer.[15]

When a problem in the user interface system arose, a mathematically competent programmer would need to be consulted. The programmer would set about adjusting not only the numerical codes that had been punched onto the paper tape, but would also adjust the settings on the computer itself.[16] Finally, depending on the unique mathematical calculations of the particular task at hand, the most favorable numerical settings could be discerned. The programmer would make the necessary alterations in the mathematical computations, again

---

12. *See generally* MICHAEL R. WILLIAMS, A HISTORY OF COMPUTING TECHNOLOGY (1985). In 1950, the prevailing criterion of system quality was performance. In the 1980s, the criterion became user friendliness and making the system easy to use. Engineers no longer use such terms as "idiot-proof" to describe user interface technology. Walter S. Mossberg, *Computers Are Still Too Complicated, But Changes Are Coming*, WALL ST. J., Oct. 23, 1997, at B1; Quick, *supra* note 11, at R14; *see also All Users Are Not Equal: Accommodating For Individual Differences In Designing Information Retrieval Systems*, (visited Mar. 3, 1998) <http://www.sisc.se/~kia/ir_individual.html> ("Our aim . . . is to look at difference [sic] designs of the information retrieval functionality together with the interface design to the system, in order to identify combinations that will meet the individual differences in users, as well as their different goals and search behavior. Potentially, this means that we have to create an interface that will actively adapt to the users, or in some other way compensate for the differences.").

13. *See* Apple Computer, Inc. v. Microsoft Corp., 799 F. Supp. 1006, 24 U.S.P.Q.2d (BNA) 1081 (N.D. Cal. 1992), *aff'd*, 35 F.3d 1435, 32 U.S.P.Q.2d 1086 (9th Cir. 1994).

14. I. BERNARD COHEN, A HISTORY OF SCIENTIFIC COMPUTING 45 (Stephen Nash ed. 1990) [hereinafter SCIENTIFIC COMPUTING].

15. *Id.* at 45.

16. *Id.* The computer, named the IBM "Automatic Sequenced Control Calculator" or "Mark I" for short, was automatic in the sense that it could perform a myriad of calculations without human intervention.

punch the codes onto the paper tape that controlled the operation, adjust the switches on the computer, and the user could once again continue the quest to receive some output.[17]

Obviously, significant developments in both software and hardware have occurred since the 1940s when paper tape represented computer user interfaces.[18] The machine itself, along with the software technology that drives it, have made monumental improvements. Of primary importance is the accessibility of computers; today's prices allow a "PC on every desk and in every home."[19] Formerly, the economics of the situation was such that only a very select group of businesses could actually afford to purchase a computer.[20] Second, the first computers demanded significant space considerations. The size of the computers were prohibitive in that individual users or even small businesses could hardly dedicate an entire room for a main frame computer.[21]

The software industry made significant progress from the paper tape days as well. Software developers who largely worked out of their garages emerged, and the industry as a whole grew to become a highly competitive, sophisticated, and lucrative industry.[22] A key to this re-

---

17. *Id.* The paper tape as a user interface device does not provide an indication of the source of the product. Therefore, protection under the Lanham Act would not be appropriate for a paper tape user interface. 15 U.S.C. § 1127 (1994).

18. *See* SCIENTIFIC COMPUTING, *supra* note 14, at 47 (noting the Mark I was operational from 1944 to 1959).

19. Bill Gates predicted that in the mid 1980s, computers would proliferate society to this extent. And, each computer, envisioned Gates, would be running Microsoft software. Transcript, *supra* note 6, at Part II.

20. *Compare* Evan Ramstad, *Low Cost PC's Made Gains Over Holiday*, WALL ST. J., Dec. 31, 1997, at A3 (personal computers are generally available for under $1000) *and* Dean Takahashi, *Envisioning the Era of the $500 PC*, WALL ST. J. NOV. 18, 1997, at B1 (some say "the next big thing in personal computing is the $500 personal computer), *with* HARRY WULFORST, BREAKTHROUGH TO THE COMPUTER AGE 174-75 (1982) (the first general purpose commercial computer was produced by International Business Machines Corp. in 1953 and leased for $20,000 per month).

21. For example, the Mark I, IBM's first computer, was 51 feet long and eight feet high. MICHAEL R. WILLIAMS, A HISTORY OF COMPUTING TECHNOLOGY 246 (1985).

22. For example, Sun Microsystems Inc. nearly doubled its net income from 1996 – 1997 and sales were $2.54 billion for the fiscal year ending June 30, 1997. Lee Gomes, *Sun Microsystems Profit, Sales Beat Wall Street's Already Bullish Forecasts*, WALL ST. J., July 18, 1977, at B6; *see also* Greg Ip., *Broad Market Rests on Narrow Strength*, WALL ST. J., Jan. 13, 1997, at C1 ("Just three stocks—Microsoft, Intel and Cisco Systems—contributed about 28% of the Nasdaq Composite Index's 22.7% gain in 1996. And IBM alone accounted for... 14% of the 1,331 increase in the Dow Jones Industrial Average which rose 26% for the year."); David Bank, *Microsoft's Problem Is What Many Firms Just Wish They Had: The Biggest Software Maker Accumulates Cash Faster Than It Can Spend It*, WALL ST. J., Jan. 17, 1997, at A3 ("Microsoft Corp[oration] faces a quandary that many companies would like to

markable growth was the recognition of the role of the user. "User friendliness"[23] became the central focus of software developers as new companies raced to provide software for a growing number of computer applications.[24]

B. *The Breakthrough*

The graphical user interfaces that exist on most people's computers today can be attributed to a single source,[25] and one that, quite ironically, does not have its name associated with computer software. In the early 1970s, a group of engineers in a Xerox Corporation research laboratory created a graphical user interface that would change the personal computer industry.[26] The achievement of these employees in develop-

---

share: It is accumulating cash far faster than it can spend it.").

23. User friendliness became a term used to describe the ease with which a computer user could interact with the machine. EDMOND H. WEISS, HOW TO WRITE USABLE DOCUMENTATION 24-25 (2d ed. 1991).

24. *Id.*

25. Microsoft's Windows operating system is run on "more than 90% of the world's personal computer's." John Wilke, *Microsoft Chief Concedes Hardball Tactics*, WALL ST. J., March 4, 1998, at B1. Antitrust issues are, however, beyond the scope of this Comment.

26. In 1971, the Xerox Corporation set up the Palo Alto Research Center (PARC) just south of San Francisco. The Xerox management believed that if people started reading computer screens instead of paper, Xerox was in trouble, so they created PARC to create the technology of the future. In 1973, Xerox built the Alto computer, which some would argue is the first personal computer. The Alto had the first graphical user interface(GUI), which allowed the user to point to information on the screen with a mouse. Xerox never attempted to market the unparalleled GUI that the researchers at PARC developed. John Warnock, former PARC researcher, explains the breakdown between developing technology and taking the technology to market:

> [T]he real frustrating part of [the whole job] was you were talking to people who didn't understand the vision and yet the vision was getting created everyday within the Palo Alto Research Center and there was not one [in Xerox Corporate headquarters in New York] to receive that vision .... When the [idea] was presented to them they had no mechanisms for turning those ideas into real live products.

Transcript, *supra* note 6, at Part III.

Meanwhile, Steve Jobs had co-founded Apple Computer in 1976 and had successfully marketed the Apple 2, the first popular personal computer. In 1979, at the peak of Apple's early success, Steve Jobs accepted a "privileged invitation" to tour Xerox's PARC. Steve Jobs described his experience while visiting PARC:

> I was so blinded by the first thing they showed me which was the graphical user interface. I thought it was the best thing I'd ever seen in my life... the germ of the idea was there and they'd done it very well and within you know ten minutes it was obvious to me that all computers would work like this some day.

Id. Jobs left Xerox that day and returned shortly thereafter with his entire programming team. Adele Goldberg, then a PARC Researcher, describes the events upon Jobs' return:

> [Steve Jobs] came back and... demanded that his entire programming team get a demo of the Smalltalk System... and I said, "No way." I had a big argument with

ing the graphical user interface system, and ultimately the failure of Xerox to take steps to protect the technology,[27] revolutionized the computer industry.[28] Consequently, Xerox's graphical user interface system, which is the predecessor to virtually all of the current software products that consumers have grown to know and love, have been borrowed/applied/taken (depending on your perspective) by other, more market-oriented competitors.[29] The rest is history.[30]

Or is it? Since the first graphical user interface system existed in 1976, *some* strides to improve graphical user interface systems have occurred, but nothing remarkable. Certainly no improvements have been made that can be truly called a milestone. To be sure, the media has routinely publicized products throughout this period,[31] but true technical progress is mysteriously absent. The reason for the lack of technical progress in user interfaces is that there is no incentive for companies to pour capital into building a technical product that receives little, if any, protection from intellectual property laws. Competitors, as evidenced from the Xerox / Apple / Microsoft interchange in graphical user interfaces, may be better advised to sit on the sidelines and play the waiting game.

---

these Xerox executives telling them that they were about to give away the kitchen sink and I said that I would only do it if I were ordered to do it cause then of course it would be their responsibility, and that's what they did.

Id. at Part III.

27. *See, e.g.*, id. at Part III (Xerox failed to appreciate the value of its technology, much less take the steps to market or to use the law to protect the system).

28. For example, Jeff Raikes, Vice-President of Microsoft, states that "[i]t really clicked with Bill [Gates] that... graphic user interface was going to be the way, the way of the future." *Id.* at Part III.

29. It should be noted that my intent is not to disparage Microsoft Corporation, Xerox Corporation, Apple Computers, Inc., their officers or employees, or any other of the players in this remarkable drama. Certainly, the computer industry as any other is premised on the survival of the fittest. However, the facts of the story itself has, in my opinion, had a clear effect on the current state of affairs for the lack of protection for graphical user interface elements in computer software.

30. Jeff Raikes, Vice President, Microsoft, stated, "[in] February or March of 1984, which was just right after the Apple Macintosh had been introduced... at the point... we were firmly convinced that we needed to bet on graphic user interface." Transcript, *supra* note 6, at Part III. John Sculley, President Apple Computer 1983-1993 states, "The look and feel which is how it looks, the experience of using it was not patentable but it was copyrightable but there was no precedent law. This was going to be a precedent setting case." *Id.* Bill Gates, "We assumed that the lawyers, the judges would all come to the right conclusion, which eventually they did." *Id.*

31. For example, Apple's launching of Macintosh on January 24, 1984 coincided with a 15 million dollar advertising campaign. *Id.* at Part III. On August 24, 1995, Microsoft launched Windows 95. The promotional campaign cost $300 million. *Id.*

The failure of the law to protect these so-called look and feel elements[32] of computer software need not be perpetuated, however. Trade dress, a subset of trademark law, is entirely applicable to graphical user interfaces because many features of computer software graphical user interfaces have distinguishing characteristics that indicate the source or origin of the product.

It is illustrative to recount how the user would interact with the computer before graphical user interfaces, including the paper tape described above along with subsequent systems; none of which functioned to indicate the source or origin of the software. For example, a computer user would type in the command "copy c, colon, backslash, document name, dot, doc, space, a, colon, backslash, return" [33] in order to save a file with a system that does not use a graphical user interface. Now however, the pictures, the arrangement of the icons, the icon selection themselves, and the pull down menus interact in a way that is much different from the generic appearance of paper tape or a black screen with white letters appearing on it. Graphical user interfaces are styled and created to interact with users and serve as a source indicating feature of the software product.

C. So What is the Problem?

There are several reasons why graphical user interfaces are crucial to the success of software. First, users want the computer program to electronically perform common functions such as editing text or the addition of numbers as naturally as when they perform the function by hand. Second, user interface elements determine the commercial success of a product as the number of individuals in the workforce use computers increases. Because there is such a high demand for computer-literate workers, employers are forced to hire less technically proficient employees. User interface elements allow a less technically inclined individual to perform in the workplace. Third, the user interface system accounts for forty percent or more of the computer program itself in a word processing or spreadsheet application, and therefore accounts for a significant portion of development costs. Fourth, peripheral or "add-on" products are often programmed to interface with

32. *See, e.g.*, Kellner, *supra* note 10, at 1011; Engineering Dynamics Inc. v. Structural Software, Inc., 785 F. Supp. 576, 580, 21 U.S.P.Q.2d (BNA) 1524, 1526 (E.D. Louisiana 1991).

33. This command illustrates how a user saves a document using DOS and is sometimes called "command line interface."

graphical user interface elements. These accessory products allow the consumer to have more choices related to their computer programs. Finally, the productivity that a worker can achieve through use of the program, and consequently the competitiveness of the product, depends largely on graphical user interface elements because employers are eager to keep employees productive and to reduce the learning curve.[34]

The importance of graphical user interface elements is highlighted when the investment in research and development is considered. Conversely, on the other end of the market, software companies are involved in extensive litigation that focuses solely on the infringement of user interface devices of computer programs.[35] The amount of attention attributed to graphical user interface elements is instructive, and there can be no debate as to the importance and value of them to both consumers and software producers.

D. Current Methods of Protecting Graphical User Interface Elements

There has been great discussion in both the legal and software communities regarding the applicable protection scheme for computer software, and whether the graphical display on a computer monitor even warrants protection.[36] In particular, courts and commentators

---

34. *See* Curtis, *supra* note 5, at 52-53 (1989).

35. *See, e.g.*, Apple Computer, Inc. v. Microsoft Corp., 35 F.3d 1435, 32 U.S.P.Q.2d (BNA) 1086 (9th Cir. 1994); Engineering Dynamics, Inc. v. Structural Software, Inc., 26 F.3d 1335, 31 U.S.P.Q.2d (BNA) 1641 (5th Cir. 1994); Apple Computer, Inc. v. Microsoft Corp., 821 F. Supp. 616, 27 U.S.P.Q.2d (BNA) 1081 (N.D. Cal. 1993); Apple Computer, Inc. v. Microsoft Corp., 759 F. Supp. 1444, 18 U.S.P.Q.2d (BNA) 1097 (N.D. Cal. 1991); Lotus Dev. Corp. v. Paperback Software Int'l, 740 F. Supp. 37, 15 U.S.P.Q.2d (BNA) 1577 (D. Mass. 1990); Digital Communications Assocs., Inc. v. Softklone Distrib. Corp., 659 F. Supp. 449, 2 U.S.P.Q.2d (BNA) 1385 (N.D. GA 1987); Broderbund Software, Inc. v. Unison World, Inc., 648 F. Supp. 1127, 231 U.S.P.Q. (BNA) 700 (N.D. Cal. 1986).

36. *See, e.g.*, Apple Computer, Inc. v. Microsoft Corp., 799 F. Supp. 1006, 24 U.S.P.Q.2d 1081 (N.D. Cal 1992) (the court held that Apple's display elements were functional and therefore not protectible under copyright laws); Broderbund Software, Inc. v. Unison World, Inc., 648 F. Supp. 1127, 1128, 231 U.S.P.Q. (BNA) 700, 701 (N.D. Cal 1986) (screen displays are protected by copyright on computer program); Digital Communications Assocs. v. Softklone Distrib. Corp., 659 F. Supp. 449, 2 U.S.P.Q.2d (BNA) 1385 (N.D. Ga. 1987) (Defendant, acting on the advice of it's legal counsel, created a software program that displayed a status screen that was identical to that of the plaintiff's. Counsel concluded that the status screen, a visual depiction of the underlying source code, was not copyrightable, although counsel acknowledged that both the source and object codes were copyrightable); Tom W. Bell, *Virtual Trade Dress: A Very Real Problem*, 56 MD. L. REV. 384 (1997) (arguing that trade dress law is applied too broadly and should be restricted); Liebman & Frishchling, *supra* note 7 (examination of trademark law's protection of hardware and software); Mitchell Zimmerman, *Trade Dress Protection for User Interfaces Revisited and Interred?*, 13 NO. 2

have had difficulty in consistently applying a body of law to user interface elements because limitations exist in both copyright and patent law when they are applied to graphical user interface elements. The following discussion highlights the shortcomings of copyright and patent protection.

1. Copyright Law

Copyright law does not adequately protect graphical user interfaces because of the relationship between the images portrayed on the screen and the underlying computer codes that create the images. Although it is arguably true that the underlying computer code is a "work fixed in a tangible medium of expression,"[37] the unique characteristics of software[38] make this unilateral directive difficult to actually apply.

Congress has expressly spoken on the issue of protection of computer software, concluding that software programs are literary works[39] under the copyright laws, proclaiming that "[o]riginal computer programs are works of authorship protected by copyright whether they are in high level computer language or machine language."[40] At face value, this scheme may appear workable, but under further analysis the shortcomings of copyright law become apparent.

COMPUTER LAW. 4 (1996) (arguing that user interface protection under trade dress is threatened).

37. 17 U.S.C. § 102 (1994).

38. *See infra* Part II for discussion on the unique characteristics of computer software that make protection of it difficult.

39. Congress created the Commission on New Technological Use of Copyrighted Works ("CONTU") to study the use and reproductions of copyrighted works, and to make recommendations to Congress regarding desirable copyright law revisions. In 1980, Congress accepted CONTU's recommendations and amended the Copyright Act almost verbatim from the CONTU report's proposed statutory changes. *See* Final Report of the National Commission on New Technological Uses of Copyright Works 3-4 (1979). __ PL __, stats. Codified at 17 USC 101 (1976). Section 101 of the Copyright Act of 1976 defines a computer program as a "set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result." 17 U.S.C. § 101 (1976).

40. Computer programs are written in specialized alphanumeric languages or "source code." In order to operate a computer, source code must be translated into computer readable form, or "object code." Object code uses only two symbols, 0 and 1, in combinations which represent the alphanumeric characters of the source code. A program written in source code is translated into object code using a computer program called an "assembler" or "compiler," then imprinted into a silicon chip for commercial distribution." Sega Enters. Ltd. v. Accolade, Inc., 977 F.2d 1510, 1514, n.2, 24 U.S.P.Q.2d (BNA) 1561, 1564, n.2 (9th Cir. 1992). Both source code and object code have been held to be protectable under the 1976 Copyright Act. American Dental Assocs. v. Delta Dental Plans Assocs., 126 F.3d 977, 978, 44 U.S.P.Q.2d (BNA) 1296, 1297 (7th Cir. 1997); Williams Elecs., Inc. v. Artic Int'l, Inc., 685 F.2d 870, 876, 215 U.S.P.Q. (BNA) 405, 410, 411 n.7 (3d Cir. 1982).

The major shortcoming in applying copyright law to graphical user interfaces is that no substantial similarity may exist between two software programs when the literal elements are compared.[41] In other words, the images on the screen display can be the result of several different underlying program codes. Accordingly, a scheme that merely protects the literal program code and either disregards or strips so-called unprotectable elements of the program from the whole is ineffectual.

Although it may be true that *some* software producers may achieve their protection goals when registering their software as a literary work, the copyright scheme as a whole has limitations when the literary work creates a picture. Referring again to the copyright statute, it might appear that the dilemma can be resolved and protection achieved through the application of the copyright provisions that enable a pictorial and graphic screen display to receive protection as an "audiovisual work."[42] Providing that the graphical display meets the ordinary standard of original, creative authorship under the statute,[43] protection may be possible.

There is one "catch," however. The Copyright Office has a one registration per work policy,[44] and when the work contains authorship elements that fall into two or more classes, the application should be filed according to the authorship class that predominates.[45] This means,

---

41. Digital Communications Assocs., Inc. v. Softklone Distrib. Corp., 659 F. Supp. 449, 456, 2 U.S.P.Q.2d (BNA) 1385, 1389 (N.D. Ga. 1987) ("[M]any different programs can create the same screen display.").

42. 17 U.S.C. §§ 101-102 (1994). Graphic displays have been copyrighted as audiovisual works. *See, e.g.*, Atari Games Corp. v. Oman, 888 F.2d 878, 12 U.S.P.Q.2d (BNA) 1791 (D.C. Cir. 1989); Red Baron-Franklin Park, Inc. v. Taito Corp., 883 F.2d 275, 11 U.S.P.Q.2d (BNA) 1548 (4th Cir. 1989); M. Kramer Mfg. Co., Inc. v. Andrews, 783 F.2d 421, 228 U.S.P.Q. (BNA) 705, (4th Cir. 1986); Williams Elecs., Inc. v. Artic Int'l, Inc., 685 F.2d 870, 215 U.S.P.Q. (BNA) 405 (3d Cir. 1982); Stern Elecs., Inc. v. Kaufman, 669 F.2d 852, 213 U.S.P.Q. (BNA) 443 (2d Cir. 1982).

43. 17 U.S.C. § 102 (1994).

44. 37 C.F.R. § 202.3(b)(6) (1998); *see also* 53 Fed. Reg. 21817. The Copyright Office expressly states that:

> [U]nder present practices . . . the Office does not register separately textual screen displays, reasoning that there is no authorship in ideas, or the format, layout or arrangement of text on the screen, and that any *literary* authorship in the screen display would presumably be covered by the underlying computer program—itself a literary work.

Copyright Office, 53 Fed. Reg. 21,817, 28,818 (1988) (to be codified at 37 C.F.R. § 202.3(b)(6)).

45. Claims to Copyright, 37 C.F.R. § 202.3(b)(2) (1998). The Copyright Office states

essentially, that software manufacturers must limit the work to one registration and hope that this sole registration provides comprehensive protection for all aspects of their multi-dimensional product.

Another flaw in the copyright regime, as applied to computer software, becomes apparent when a court needs to determine whether another work is infringes the original work. In order to discern whether two works are substantially similar, the courts undertake either an abstraction-filtration-comparison analysis[46] or similar method,[47]neither of which is effective in analyzing graphical user interfaces because the value of them is in the whole.

Because of these glaring weaknesses in copyright law as applied to graphical user interface elements, courts, not surprisingly, have been unable to offer any cogent principles to use when analyzing whether one graphical user interface infringes on another. A review of court decisions on this topic reveals that judges have had difficulty applying the copyright law to computer software generally, and specifically to graphical user interface elements. For example, some courts have held that protection of computer programs under copyright law extends only to the literal portion of the software, which is the source and object code.[48] Other courts have protected the literal elements, including the sequence and organization of the program, in addition to the text and artwork of the audiovisual display.[49] And yet a third analysis allowed

---

that "[i]n considering the issue of computer screen displays . . . there is no sound basis for departing from the principles of these regulations in the case of computer programs and related screens." Copyright Office 53 Fed. Reg. 21,817, 21,819 (1988). The Copyright Office goes on to support this policy choice with the reasoning that the public record will be better reflected which benefits all parties, including copyright claimants, the public, and the Copyright Office. *Id.*

46. *See* Lisa M. Gable, Note, *The Feasibility of the Abstraction-Filtration-Comparison Test for Computer Software Copyrightability (And Analysis of* Bateman v. Mnemonics*)*, 14 GA. ST. U. L. REV. 447 (1998).

47. *Compare* Computer Assocs. Int'l v. Altai, 982 F.2d 693, (2d Cir. 1992), *with* Lotus Dev. Corp. v. Paperback Software Int'l, 740 F. Supp. 37, 15 U.S.P.Q.2d (BNA) 1577 (D. Mass. 1990).

48. *See, e.g.*, Synercom Tech., Inc. v. University Computing Co., 462 F. Supp. 1003, 199 U.S.P.Q. (BNA) 537 (N.D. Tex. 1978); *see generally* Marci A. Hamilton *Computer Science Concepts in Copyright Cases: The Path to a Coherent Law*, 10 HARV. J.L. & TECH. 239 (1997).

49. *See* Whelan Assocs., Inc. v. Jaslow Dental Lab., Inc., 797 F.2d 1222, 230 U.S.P.Q. (BNA) 481 (3d Cir. 1986) (holding that "copyright protection of computer programs may extend beyond the programs' literal code to their structure, sequence, and organization . . . ."); Broderbund Software, Inc. v. Unison World, Inc., 648 F. Supp. 1127, 231 U.S.P.Q. (BNA) 700 (N.D. Cal. 1985) (determining that "the overall structure, sequence, and arrangement of the screens, text, and artwork . . . are protected under copyright laws . . . .").

copyright protection in a screen based on the "compilation"[50] of the menu terms, although the court explicitly stated that copyright in a computer program does not extend to the screen displays.[51]

The fundamental problem with copyright protection for software is the distinction between ideas[52] and the expression of ideas.[53] Courts are getting caught up in an attempt to "peel" [54] the program into minute components instead of viewing the program as a composite whole.[55] Because this approach fails to recognize that the value of the program is the graphical display in its entirety and how the consumer views the product, the user interface should be neither peeled nor otherwise carved up. Rather, the software graphical display should remain intact for purposes of an infringement analysis. Important decisions have illustrated this shortcoming.

---

50. 17 U.S.C. § 103 (1994).

51. Digital Communications Assocs., Inc. v. Softklone Distrib. Corp., 659 F. Supp. 449, 445, 2 U.S.P.Q.2d (BNA) 1385, 1388 (N.D. Ga. 1987).

52. *Whelan*, 797 F.2d at 1234-37, 230 U.S.P.Q. (BNA) at 490 ("[T]he purpose ... of [the] ... work [is] the work's idea, and everything that is not necessary to that purpose ... [is] part of the expression of the idea."); Ashton-Tate Corp. v. Ross, 728 F. Supp. 597, 601-02, 12 U.S.P.Q.2d (BNA) 1734, 1737 (N.D. Cal. 1989) (holding that the "[l]ist of [user] commands is only an idea that is not protected under federal [copyright] law.").

53. Granting a monopoly and allowing an individual to protect the idea itself would run counter to the purpose of Copyright Act. Baker v. Selden, 101 U.S. 99 (1879) (holding that an accounting system was not copyrightable because it was an idea).

54. *See* Brown Bag Software v. Symantec Corp., 960 F.2d 1465, 1475-76, 22 U.S.P.Q.2d (BNA) 1429, 1437 (9th Cir. 1992) (concluding that the purpose of analytic dissection is to define scope of copyright protection); *see also* Gable, *supra* note 46 (discussing the application of the abstraction-filtration-comparison analysis to computer software).

55. When courts consider software there has been general confusion as to when to focus on the program as it exists as a whole and when to focus on individual elements. *See, e.g.*, Engineering Dynamics, Inc. v. Structural Software, Inc., 26 F.3d 1335,1342-43 (5th Cir. 1994); *see generally* David W.T. Daniels, Comment, *Learned Hand Never Played Nintendo: A Better Way to Think About the Non-literal, Non-Visual Software Copyright Cases*, 61 U. CHI. L. REV. 613 (1994) (noting that there is precedent for viewing other products in their entirety); *see also* Dr. Seuss Enters., L.P. v. Penguin Books USA, Inc., 109 F.3d 1394, 1398-1402,42 U.S.P.Q.2d 1184, 1187-90 (9th Cir. 1997) (noting that the Ninth Circuit modifies the *Kroft* test to use a subjective and objective inquiry); Sid & Marty Kroft Television Prods., Inc. v. McDonald's Corp., 562 F.2d 1157, 1167-68, 196 U.S.P.Q. (BNA) 97, 105-106 n.9 (9th Cir. 1977) (holding that defendant McDonald's characters infringed the "total concept and feel" of Plaintiff's "Pufnstuf" characters). The Ninth Circuit uses a two-part test for substantial similarity. First, the extrinsic test inquires whether there is a similarity of ideas; in which the court finds "analytical dissection" appropriate. Second, part two of the test asks whether an ordinary and reasonable person would perceive a substantial taking of the protected expression. *Id.* at 1168, 196 U.S.P.Q. (BNA) at 106; Roth Greeting Cards v. United Card Co., 429 F.2d 1106, 1110, 166 U.S.P.Q. (BNA) 291, 294 (9th Cir. 1970) (suggesting that questions of infringement for greeting cards should be viewed as a whole).

The court in *Apple Computer, Inc. v. Microsoft Corp.*,[56] for example, accepted the premise that Apple's software had a particular look and feel to it, but the court declined to accept Apple's argument that the manner in which the screen is organized into a "desktop metaphor"[57] is an expression of an idea.[58] The court described the graphical images on the screen as functional elements of the program[59] and denied copyright protection of its "look and feel" because the visual display was so closely tied to the functional purpose of the article that the display becomes "standard."[60] Citing "market factors" in reference to the expectations of users, the effect of the court's decision was to allow Apple's expression of its idea to be used by other software producers.[61]

The difficulty that courts have experienced, including the court in *Apple*, is the inability to clearly and consistently articulate the difference between an idea and its expression. The inability to recognize this difference prohibits software producers from protecting potentially landmark developments in advanced user interface concepts that Xerox created,[62] that Apple subsequently used as a conceptual model,[63] and that today's software developers repeatedly use. The blatant attempt by Congress to pound a square peg (all computer software) into a round hole (the copyright law) is clearly ineffectual.

2. Patent Law

The last word from the Supreme Court is that computer software is considered appropriate subject matter to receive patent protection.[64]

---

56. 799 F. Supp. 1006, 24 U.S.P.Q.2d (BNA) 1081 (N.D. Cal. 1992).

57. The term "desktop metaphor" is a term given to a graphical user interface system that roughly resembles a desktop with accessories such as scissors, paste, file folders, and others.

58. *Id.* at 1022-23, 24 U.S.P.Q.2d (BNA) at 1088.

59. *Id.*

60. *Id.* at 1023, 24 U.S.P.Q.2d (BNA) at 1089.

61. *Id*; *see also* Randall M. Whitmeyer, *A Plea For Due Processes: Defining The Proper Scope Of Patent Protection for Computer Software*, 85 NW. U. L. REV. 1103 (1991) (providing an in depth treatment of the construction of computer software).

62. *See generally* Dr. David Canfield Smith et al., *Designing the Star User Interface*, 7 BYTE 242 (Apr. 1982); Jeff Johnson et al., *The Xerox Star: A Retrospective*, 22 COMPUTER 11 (Sept. 1989).

63. Chris Morgan, *The Lisa Computer System*, 8 BYTE 90,108 (Feb. 1983) ("we made some changes to our user interface system based on ideas we got from [the Xerox Star].").

64. Diamond v. Diehr, 450 U.S. 175, 209 U.S.P.Q. (BNA) 1 (1981) (Supreme Court held that computer program containing mathematical algorithm that controlled a machine that cured synthetic rubber was patentable). *But see* Gottschalk v. Benson, 409 U.S. 63, 175 U.S.P.Q.2d (BNA) 673 (1972) (no protection for method of converting binary-coded deci-

Although the question of whether an entire graphical user interfaces warrant patent protection has not yet been raised directly, computer software and even some of its graphical components[65] have been registered as patents.

A patent grants to its owner a monopoly for worthy development and allows the owner the right to exclude competitors from copying or otherwise using the patented object.[66] Since competition is precluded for a limited duration, the patent holder is usually successful in achieving economic rewards. Society benefits as well from this arrangement, because the patent holder must fully disclose the inventions so the public has the ability to improve upon the patented device.[67]

The drafters of the United States Constitution sought to encourage the growth of technology. Therefore, the nature of the products or devices that receive patent protection are those that contribute to the advancement of science and technology.[68] In recognition of this objective, the Supreme Court has articulate several purposes of the Patent Act:

> First, patent law seeks to foster and reward invention; second, it promotes disclosure of inventions, to stimulate further innovation and to permit the public to practice the invention once the patent expires; third, the stringent requirements for patent pro-

---

mals into binary numerals for use with a computer); Parker v. Flook, 437 U.S. 584, 198 U.S.P.Q.2d (BNA) 193 (1978) (no protection for software program employing abstract mathematical formula with the purpose of updating alarm limits during catalytic conversion process).

65. *See* Michael J. Schallop, Note, *Protecting User Interfaces: Not as Easy as 1-2-3*, 45 EMORY L. J. 1533, 1547-49, n.77-90 (1996) (individual graphical user interface elements for which patents have issued are noted).

66. Article I, Section 8, Clause 8 of the United States Constitution provides that Congress shall have the power "[t]o promote the Progress of Science and useful Arts, by securing for limited Times to Authors and Inventors the exclusive Right to their respective Writings and Discoveries." *Id.* Design patents have a term of 14 years, 35 U.S.C. § 173 (1994), and utility patents have a term of 20 years from the date the patent application is filed, 35 U.S.C. § 154(a)(2) (1994).

67. Kewanee Oil Co. v. Bicron Corp., 416 U.S. 470, 480-81, 181 U.S.P.Q. (BNA) 673, 678 (1947); *see also* Bonito Boats, Inc. v. Thunder Craft Boats, Inc., 489 U.S. 141, 151-156, 9 U.S.P.Q.2d (BNA) 1847, 1850-52 (1989) (explaining that a bargain is struck between the inventor and the public: the inventor freely discloses the invention and receives a monopoly for a limited duration in return. As a policy matter, this incentive advances science and technology by allowing others to improve upon the patented item while the inventor has the opportunity to benefit economically); Kevin Mohr, *At the Interface of Patent and Trademark Law: Should a Product Configuration Disclosed in a Utility Patent Ever Qualify for Trade Dress Protection?*, 19 HASTINGS COMM. ENT. L.J. 339, 347 (1997).

68. *See* 35 U.S.C. § 101 (1994) (utility requirement).

tection seek to assure that ideas in the public domain remain there for the free use of the public.[69]

Disclosure and use are the "hallmarks" of the patent system[70] and inventors must satisfy stringent conditions to obtain either of two species of patents: a utility or a design patent.[71] In order to obtain a utility patent,[72] an inventor must show that the invention is useful,[73] novel,[74]

---

69. Aronson v. Quick Point Pencil Co., 440 U.S. 257, 262, 201 U.S.P.Q. (BNA) 1, 4-5 (1979) (citing *Kewanee*, 416 U.S. at 480-81, 181 U.S.P.Q. (BNA) at 678); *see also* Graham v. John Deere Co., 383 U.S. 1, 5-6, 148 U.S.P.Q. (BNA) 459, 462 (1966). The Court states that Congress may not:

> [E]nlarge the patent monopoly without regard to the innovation, advancement or social benefit gained thereby. Moreover, Congress may not authorize the issuance of patents whose effects are to remove existent knowledge from the public domain, or to restrict free access to materials already available. Innovation, advancement, and things which add to the sum of useful knowledge are inherent requisites in a patent system which by constitutional command must 'promote the Progress of . . . [Science and the] useful Arts.

Graham, 383 U.D. at 5-6, 262 U.S.P.Q. at 462-63.

70. *Bonito Boats*, 489 U.S. at 156-57, 9 U.S.P.Q.2d (BNA) at 1854; *see also* Singer Mfg. Co. v. June Mfg. Co., 163 U.S. 169, 185 (1896).

71. Mohr, *supra* note 67, at 347.

72. *See* J.H. Reichman, *Legal Hybrids Between The Patent And Copyright Paradigms*, 94 COL. L. REV. 2432, 2456 (1994). Historically, utility patents have been employed to protect the external aspects of a common tool, not considering the system or underlying idea. *Id.* Such historical principles may enable practitioners and theorists to view the current intellectual regime as it applies to software differently.

The utility model laws are likely the forerunners of intellectual property protection as we now know them. The purpose of utility laws was largely to protect property developed within an industrial setting such as handtools and other such devices. *Id.* at 2456. Prior to the development and application of utility laws, the belief was that industrial tools were not protectible under early regimes because of their functionality. *Id.* The overriding theory of utility model laws is to protect the external aspects of the tool and not consider the system or underlying idea. *Id.* at 2456-57. The advantages of this system are many. First, the approach was holistic and accounted for the external features of the article. Second, utility laws were adept at providing protection quickly, averaging three to four months. This is applicable to German patents. *Id.* at 2457 n.9. This allows the system of protection to function at a rate approximating the pace of technical developments. *Id.* at 2456-57. These advantages are applicable to software because of the manner in which they operate and may potentially be desirable to computer software developers.

73. 35 U.S.C. § 101 (1994); *see also* JAY DRATLER, JR., INTELLECTUAL PROPERTY LAW: COMMERCIAL CREATIVE AND INDUSTRIAL PROPERTY § 2.03[2] (1996) (an inventor must show that the invention has commercial potential or other known use); Raytheon Co. v. Roper Corp., 724 F.2d 951, 959, 220 U.S.P.Q. (BNA) 592, 598 (Fed. Cir. 1983). *But see* Brenner v. Manson, 383 U.S. 519, 532-36, 148 U.S.P.Q. (BNA) 689, 693 (1966).

74. 35 U.S.C. § 102. The United States rewards the first person to invent, rather than the first person to register the invention. *Id.*; *see also Bonito Boats*, 489 U.S. at 148-49, 9 U.S.P.Q.2d (BNA) at 1851 (it is necessary that the invention be new, and not already known or in use).

and nonobvious.[75] A design patent requires a showing that the invention is ornamental,[76] novel,[77] and nonobvious.[78]

The purposes of utility and design patents are different as well. While utility patents are intended to reward inventors who contribute to the "useful arts," design patents are intended to promote the "decorative arts."[79] As applied to graphical user interfaces, the largest overlap occurs with design patents. Because the display of the screen may occur by a virtually unlimited number of ways, graphical user interface elements should not be classified as functional features that would fall within the exclusive province of utility patents.[80]

Design patents could provide some protection for graphical user interfaces; however, some limitations are apparent. Design patents require a difficult application procedure, a relatively short duration of protection,[81] and even a judiciary that some would argue is openly hostile.[82] Further, graphical user interface elements have unique characteristics including the necessity to improve the product, often incrementally and over a period of time,[83] which further makes patent protection an unsatisfactory answer.

As applied to graphical user interfaces, utility patent and trade dress protection are mutually exclusive.[84] Design patent and trade dress pro-

---

75. 35 U.S.C. § 103 (providing that the invention may not be "obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains"); *see generally* Graham v. John Deere Co., 383 U.S. 1, 11, 148 U.S.P.Q. (BNA) 459, 464-66 (1966).

76. 35 U.S.C. § 171; *see also* Power Controls Corp. v. Hybrinetics, Inc., 806 F.2d 234, 238-39, 231 U.S.P.Q. (BNA) 774-777 (Fed. Cir. 1986) (it is further required that the particular aspect for which a design patent is sought be non-functional).

77. *See supra* note 74.

78. *See supra* note 75.

79. *See* Mohr, *supra* note 67, at 359.

80. The term "functional" refers to shapes or features that have a useful or utilitarian purpose. If the feature is necessary to the product, or if the feature makes the product cheaper to make, then it will be held to be legally functional. *See, e.g., In re* Morton-Norwich Prods., Inc., 671 F.2d 1332, 1339-40, 231 U.S.P.Q. (BNA) 9, 14-15 (C.C.P.A. 1982). Functionality, discussed more comprehensively in § III(B)(3), prevents a person from obtaining perpetual protection under trademark laws when patent laws would prohibit it.

81. Mohr, *supra* note 67, at 375.

82. Fabrication Enter. v. Hygenic Corp., 64 F.3d 53, 35 U.S.P.Q.2d (BNA) 1753 (2d Cir. 1995); *see also* RAYMOND T. NIMMER, THE LAW OF COMPUTER TECHNOLOGY § 2.06(5) (2d ed. 1992) (suggesting that most computer software application programs would not meet the non-obviousness standard under section 103 of the Patent Act).

83. *See infra* Part II.B for discussion on the unique characteristics of computer software.

84. The Patent and Trademark Office will reject an application for trade dress registra-

tection, however, are not.[85] Courts and commentators are generally in agreement that design patent and trade dress protection may coexist with respect to one product[86] and does not compromise the Supreme Court's prohibition against trademark protection providing a "shortcut" to achieve perpetual utility patent protection.[87]

Patent law may provide another avenue of protection for graphical user interface elements; however, the weakness under this regime, as under copyright law, is the necessity to dissect the software program and to register individual elements of the software program. The cost to register an entire graphical user interface for one software program may prove to be prohibitive and unworkable.

II. INHERENT QUALITIES OF SOFTWARE THAT MAKES PROTECTION OF SOFTWARE DIFFICULT

Software is unique because it is multi-dimensional,[88] and consequently, application of the law does not come easily. The tension cre-

---

tion if the dress is functional, while a utility patent, by definition is useful or functional. *See, e.g., In re* Bose Corp., 772 F.2d 866, 871-72, 227 U.S.P.Q. (BNA) 1, 5-6 (Fed. Cir. 1985); *In re* Cabot Corp., 15 U.S.P.Q.2d (BNA) 1224, 1229 (T.T.A.B. 1990); *see generally* 1 J. THOMAS MCCARTHY, MCCARTHY ON TRADEMARKS AND UNFAIR COMPETITION § 7.89 (1997).

85. Another issue that is relevant, but beyond the scope of this Comment, is what a court should do when a product could be protected by a design patent but the owner instead seeks trade dress protection. *See, e.g.*, Kohler Co. v. Moen, Inc., 12 F.3d 632, 29 U.S.P.Q.2d (BNA) 1241 (7th Cir. 1993); Hubbell, Inc. v. Pass & Seymour, Inc., 883 F. Supp. 955, 35 U.S.P.Q.2d (BNA) 1760 (S.D.N.Y. 1995).

86. *See generally Kohler*, 12 F.3d at 636-43, 29 U.S.P.Q.2d (BNA) at 1244-50; W.T. Rogers Co. v. Keene, 778 F.2d 334, 337, 228 U.S.P.Q. (BNA) 145, 145-46 (7th Cir. 1985); Jay Dratler, Jr., *Trademark Protection for Industrial Designs*, 1988 U. ILL. L. REV. 887, 928-935 (1988); Mohr, *supra* note 67 at 375-80; 1 MCCARTHY, *supra* note 84, §7.90-91.

87. Qualitex Co. v. Jacobson Prods. Co., 514 U.S. 159, 164-65, 34 U.S.P.Q.2d (BNA) 1161, 1164-65 (1995) (explaining how hindrances to competition is avoided by the doctrine of functionality in trade dress law); Marketing Displays, Inc. v. Traffix Devices, Inc., 967 F. Supp. 953, 964 (E.D. Mich. 1997); *see generally* 1 MCCARTHY, *supra* note 84, §7:89, 7-203; Dratler, *supra* note 73, at 939 ("At its core, the functionality doctrine is based upon a simple premise: trademark law should not provide a shortcut to achieve perpetual utility patent protection . . . If a feature of an article is dictated by such a useful purpose, it is functional and must be protected, it at all, by a utility patent.") (citations omitted); *see also infra* Part III.B.3.

88. *See, e.g.*, Apple Computer, Inc. v. Microsoft Corp., 35 F.3d 1435, 1441, 1442-43, 32 U.S.P.Q.2d (BNA) 1086, 1091-92 (9th Cir. 1994) (affirming the district courts practice of dissecting the software program "into individual elements" instead of viewing the "total concept and feel" of the work); Brown Bag Software v. Symantec Corp., 960 F.2d 1465, 1473, 22 U.S.P.Q.2d (BNA) 1429 (9th Cir. 1992) (counting and comparing default color selections between works); Data East USA, Inc. v. Epyx, Inc., 862 F.2d 204, 208-09, 9 U.S.P.Q.2d (BNA) 1322 (9th Cir. 1988) (moving figures depicted in games are inherent in sport of karate and do not warrant finding of infringement); Apple Computer v. Microsoft Corp., 821 F. Supp. 616,

ated by new technologies and the lack of a clear "fit" is certainly not new in the law, however.[89] Nor does this tension justify the abandonment of the entire intellectual property system as it relates to computer software.[90]

Computer software has several characteristics that make it truly unique, and operate to directly affect the intellectual property protection scheme applicable to it. The characteristics that distinguish software from other products are worth noting because they directly influence the law's ability to protect software.

A. *Underlying the Visual Display is Computer Code that is Capable of Producing Several Visual Results*

Courts have seemingly struggled with applying copyright principles to computer interface elements because the visual display on a computer screen display is not necessarily a copy of any particular computer program underlying it. The court in *Digital Communications Associates, Inc. v. Softklone Distributing Corp.*[91] aptly noted:

> The distinction between programs and screen displays lies in the fact that if one has a fixed computer program, one can, with the aid of a computer, repeatedly produce the same screen display. Thus, a computer program is a copy of a screen display. The converse, however, is not true. If one has a fixed screen display, one cannot, even with the aid of a machine, repeatedly create the same program as many different programs can create the same screen display. Under the above analysis, it is clear that the

---

619, 27 U.S.P.Q.2d (BNA) 1081, 1085 (N.D. Cal. 1993) ("[A]n outline to indicate a moving window is not protectible...."); Apple Computer, Inc. v. Microsoft Corp., 759 F. Supp. 1444, 1449-50, 18 U.S.P.Q.2d (BNA) 1097, 1100 (N.D. Cal. 1991) (the court lists portions of the visual display and attempts to describe motion).

89. Technology and change in general also exists within other industries and implicates the applicability of the law. *See, e.g.,* Burnham v. Superior Court of Cal., 495 U.S. 604, 617 (1990) ("[C]hanges in technology of transportation and communication... led to an 'inevitable relaxation of the strict limits on state jurisdiction' over nonresident individuals and corporations."); Sony Corp. of Am. v. Universal City Studios, Inc., 464 U.S. 417, 430-31, 220 U.S.P.Q. (BNA) 665, 673 (1984) ("Repeatedly, as new developments have occurred in this country, it has been Congress that has fashioned the new rules that new technology made necessary."); Columbia Broad. Sys., Inc. v. Democratic Nat. Comm., 412 U.S. 94, 102 (1973) ("The problems of regulation are rendered more difficult because the broadcast industry is dynamic in terms of technological change....").

90. *See* Ginsburg, *supra* note 9, at 2559.

91. 659 F. Supp. 449, 2 U.S.P.Q.2d (BNA) 1385 (N.D. Ga. 1987).

> plaintiff must establish another basis, other than the computer program, for copyright protection of the status screen.[92]

This summarizes and correctly identifies the problem with attempting to protect graphical user interfaces under copyright law: the same graphical display can be created by a number of computer programs.[93] Consequently, it is illogical to conclude that a screen display can be a "copy" when the effect of completing different computer code can create the same display.

B. Incremental Innovations Drive Software Developments

The demand for computer software makes the industry one of the fastest growing segments of the economy.[94] Despite the prevalence of computer software and the extent to which it can be applied for personal and business uses, the industry as a whole progresses in often very small increments. A paradox of sorts exists when one considers the oftentimes minute advances in the "lifetime" of an individual software program.

Software manufacturers frequently release several editions of one program to the market.[95] Because software programs are very complex, individual elements within the whole progress in small pieces as programmers work out weaknesses to improve the product.[96] This process of successive improvements within the software arena is called incremental innovation.[97] Because sometimes minute advances in a given

---

92. *Id.* at 456, 2 U.S.P.Q.2d (BNA) at 1389.

93. *Id*; *see also* Johnson Controls, Inc. v. Phoenix Control Sys., Inc., 886 F.2d 1173, 1175-76, 12 U.S.P.Q.2d (BNA) 1566, 1569 (9th Cir. 1989) (affirming the district courts decision that nonliteral components of computer software including the structure, sequence and organization and user interface "without copying the literal elements of the program, which include the source and object code"); Computer Assocs. Int'l, Inc. v. Altai, 982 F.2d 693, 712-15, 23 U.S.P.Q.2d (BNA) 1241, 1258-60 (2d Cir. 1992) (finding no substantial similarity between plaintiff's and defendant's computer program after defendant "cleaned up" program after substantially copying plaintiff's program).

94. *See* International Trade Admin., U.S. Dept. of Commerce, U.S. Industrial Outlook 1994, at 27-1 (1994) (computer software market is one of the fastest growing sectors in the United States economy); International Trade Admin., U.S. Dept. of Commerce, U.S. Global Trade Outlook 1995-2000: *Toward the 21st Century*, at 134-35 (1995) (from 1991 to 1994, the market for packaged software in the United States increased by nearly 41%).

95. For example, Microsoft has released Windows in versions 1.0, 2.0, 2.03, 3.0, 3.1, 3.11, Windows 95 and will soon release Windows 98.

96. *Cf.* Reichman, *supra* note 72, at 2511-19.

97. Gross, *supra* note 9, at 124-25 (one software program consists of millions of numerals that make up the binary code and that results in the graphical displays that the user views. The code and the resulting graphics are improved in small portions).

program is capable of making a monumental impact in the technological world, such seemingly small but important innovations are noteworthy as well as economically valuable.[98]

The existing climate in the software industry, driven in increments, demands an intellectual property regime that can offer protection to a product that is consistent, reliable, and comprehensive as the evolution of the product occurs. Flexibility is one such characteristic of a suitable protection regime. Because successful software programs have sometimes nine lives or more, it is imperative that successful features of the software program that represent the program to consumers be protected throughout the "lifetime" of the product.[99]

C. Software Innovators are Often Interdependent

An inherent irony exists within the software industry: each innovator is at the same time a user and a provider of information.[100] Software programmers rely on the knowledge of innovators that have come before them, but the code each new program is created "from scratch" rather than the purchaser licensing or otherwise purchasing the technology.[101] The charade that courts have accepted is that the copyrighted software programs at issue in the litigation before them are original works of authorship.[102]

The mutual dependence[103] created between virtually all software innovators is a very different climate from that which exists for authors creating works with mediums such as books, musical numbers, sculp-

---

98. *See generally* Reichman, *supra* note 72, at 2527-2533 (suggesting automatic licensing for small-scale innovations).

99. *See generally* Robert P. Merges, *Intellectual Property and the Costs of Commercial Exchange: A Review Essay*, 93 MICH. L. REV. 1570 (1995) (arguing that vesting strong rights in a single party will encourage that party to make efficient use of the property).

100. The concept of interdependency of innovators closely relates to incremental innovation.

101. Protection under the Copyright Act expressly limits protection to the material of the work contributed by the author of such work, "as distinguished from the preexisting material employed in the work." 17 U.S.C. § 103 (b) (1994).

102. *See, e.g.*, Apple Computer, Inc. v. Microsoft Corp., 759 F. Supp. 1444, 1454, 18 U.S.P.Q.2d (BNA) 1097 (N.D. Cal. 1991) (copyright owner's failure to disclose to Copyright Office that designers of interface were influenced by preexisting works by Xerox Corporation and "such borrowing of ideas" does deprive Apple's works of their presumption of validity).

103. *But see* Mark Lemley, *Encouraging Software Reuse*, 49 STAN. L. REV. 255 (1997) (arguing that current protection regimes discourage the reuse of computer programs and "reinventing" is costly, wastes resources, and is prevalent in the software industry).

tures, or paintings, where most works are produced independently from one another. The implications of this inter-relatedness as it relates to the law when applied to software exposes a glaring weakness in copyright law as it relates to graphical user interfaces.

D. Reverse Engineering Gives Away the Underlying Code

Reverse engineering is the process of dismantling and examining the components of a software program to discover precisely what makes the program work.[104] By looking at the internal[105] structure of the software, a competitor can see the code that enables the software to perform as it does.[106] The second comer, in economic terms, incurs little of the costs of producing the product[107] and is in effect a free-rider, preying upon the research, development, and talents of the product's innovators.[108] The drafters of the CONTU Final Report[109] recognized the fact that the cost

---

104. *See* Marshall Leaffer, *Engineering Competitive Policy And Copyright Misuse*, 19 U. DAYTON L. REV. 1087, 1089 (1994); John G. Mills, *Possible Defenses to Complaints for Copyright Infringement and Reverse Engineering of Computer Software: Implications for Antitrust and I.P. Law,* 80 J. PAT. & TRADEMARK OFF. SOC'Y 101 (1998).

105. *See* J.H. Reichman, *Electronic Information Tools - The Outer Edge of World Intellectual Property Law*, 24 INT'L REV. INDUS. PROP. & COPYRIGHT L. 446, 472 (1993). Rod Canion, Co-founder of Compaq, also stated that, "[r]everse engineering is figuring out after something has already been created how it ticks, what makes it work, usually for the purpose of creating something that works the same way or at least does something like the thing you're trying to reverse engineer." Transcript, *supra* note 6, at Part II.

106. Compaq sought to compete with IBM in the personal computer market. In so doing, it was necessary to copy the ROM-BIOS, a vital chip in the computer that connected the hardware with the software. Rod Canion, Co-founder, Compaq, states: "First you have to decide how the ROM works, so what we had to do was have an engineer sit down with that code and through trial and error write a specification that said here's how the BIOS ROM needs to work. It couldn't be close; it had to be exact[.]" *Id.* at 12.

107. Rod Canion, Co-founder, Compaq, reported that Compaq dedicated 15 senior programmers for several months at a cost of $1 million to reverse engineer IBM's personal computer, and states "In our first year of sales, we set an American business record. I guess maybe a world business record. Largest first year sales in history. It was a hundred and eleven million dollars." *Id.* at 13.

108. William M. Landes & Richard A. Posner, *Trademark Law: An Economic Perspective*, 30 J.L. & ECON. 265, 270 (1987). Reverse engineering is not always the most efficient means of arriving at a solution. Other solutions such as direct licensing or the application of one's own resources, including existing skills and knowledge, may prove to be lest costly. In the computer software industry; however, reverse engineering is a significant problem because the medium itself lends to rapid and low cost reproduction, while costs to develop the product in the first place is significant, and may compel a party to abandon the innovative route. *Id.; see also* Wendy J. Gordon, *An Inquiry Into the Merits of Copyright: The Challenges of Consistency, Consent, and Encouragement Theory*, 41 STAN. L. REV. 1343 (1989).

109. *See supra* note 39.

of developing computer programs is considerably greater than the cost of duplicating a program.[110]

Despite potential objections from an ethical standpoint, reverse engineering holds an important position in the computer industry and after the holdings in *Atari Games Corp. v. Nintendo of America, Inc.*[111] and *Sega Enterprises, Ltd. v. Accolade, Inc.*,[112] reverse engineering is a fair use when used for the purpose of developing non-infringing or compatible software.[113]

The court in *Accolade* examined the nature and the characteristics of Accolade's direct use of the copied works, the use to which Accolade put the functional information it obtained, and the nature of the market, and concluded that Accolade's use of the information is a fair use.[114] This result allows competitors, along with anyone else, to discover the features and systems that make the program work.

Reverse engineering holds an important role in software development.[115] Competing interests and public policy considerations need to be balanced in relation to the interests of the author of the computer software program.

E. Software Code is a System

Computer software consists of a set of instructions. The instructions, or code, is protected under copyright laws from literal copying. The problem that arises, however, is that some source code is so standard and so basic that it is universal to many software programs and therefore is necessary for the most elementary of programs.[116] Under

---

110. CONTU Report at 11.

111. 975 F.2d 832, 843, 24 U.S.P.Q.2d (BNA) 1015, 1023 (Fed. Cir. 1992).

112. 785 F. Supp. 1392, 23 U.S.P.Q.2d (BNA) 1440 (N.D. Cal. 1992), *aff'd* in part, *rev'd* in part, 977 F.2d 1510 (9th Cir. 1993).

113. Sega Enters. Ltd. v. Accolade, Inc., 977 F.2d 1510, 1527-28 (9th Cir. 1992) ("[As a matter of law,] when disassembly is the only way to gain access to ideas and functionality . . . disassembly is fair use."); Sega Enters. Ltd. v. Accolade, Inc., 785 F. Supp. 1392, 1399, 23 U.S.P.Q.2d 1440, 1455 (N.D. Cal. 1992).

114. 977 F.2d 1510, 1520, 24 U.S.P.Q.2d (BNA) 1561, 1568 (9th Cir. 1992).

115. Software programmers have attempted to prevent copying and reverse engineering by programming impediments into their software programs. However, specific software programs that decode or circumvent these protectionist measures are highly successful at doing so. And so the cycle continues. *See generally* Lemley, *supra* note 103, at 263-66.

116. *See, e.g.*, Telemarketing Resources v. Symantec Corp., 12 U.S.P.Q.2d (BNA) 1991,1995 (N.D. Cal. 1989), *aff'd* in part, *vacated* in part, *remanded* by, Brown Bag Software v. Symantec Corp., 960 F.2d 1465, 22 U.S.P.Q.2d 1429 (9th Cir. 1992) (some concepts and operations are fundamental to a host of computer programs and are therefore unprotectable);

the copyright law, a provision is made for this characteristic; Congress states that "systems" may not be protected under copyright law for just this reason.[117] Courts have generally followed this mandate and have rightfully allowed systems to remain in the public domain because virtually every programmer needs access to them.[118] However, where these basic and standard codes end and where protected software begins remains unclear. Courts have not been able to draw this line and they have been unclear regarding what constitutes basic program code from that which is protectible.[119] In practical terms, some courts have allowed a competitor to copy the visual result of code if the program can be written in a manner that does not duplicate the literal expression.[120] This provides a window of opportunity that allows capitalization of a successful user interface element.

Because computer software is a unique form of media, the law should be structured so that the unique and particular characteristics of computer software may be considered in the law's application. Although copyright and patent laws are applicable to computer software in some circumstances and for some software products as a whole, the shortcomings of these regimes are evident. Trade dress law, however,

---

see also Apple Computers, Inc. v Microsoft Corp., 799 F. Supp. 1006, 1026, 24 U.S.P.Q.2d (BNA) 1081, 1092 (N.D. Cal. 1992) (Apple's desktop interface along with the accompanying features are common to all the works involved in lawsuit and are not protectable).

117. *See* 17 U.S.C. § 102(b) ("In no case does copyright protection for an original work of authorship extend to any idea, procedure, process, system ... regardless of the form in which it is described ....").

118. Whelan Assocs., Inc. v. Jaslow Dental Lab., Inc., 797 F.2d 1222, 1248, 230 U.S.P.Q. (BNA) 481, 500 (3d Cir. 1986) ("copyright protection of computer programs may extend beyond the programs literal [source and object] codes to their structure, sequence and organization ....").

119. *See, e.g.,* Williams Elecs., Inc. v. Artic Int'l, Inc., 685 F.2d 870, 215 U.S.P.Q. (BNA) 405 (3d Cir. 1982); Stern Elec., Inc. v. Kaufman, 669 F.2d 852, 855, 213 U.S.P.Q. (BNA) 443, 445 (2d Cir. 1982) ("many different computer programs can produce the same 'results ....'"); Digital Communications Assocs., Inc. v. Softklone Distrib. Corp., 659 F. Supp. 449, 455-56, 2 U.S.P.Q.2d (BNA) 1385, 1388 (N.D. Ga. 1987) ("Screen displays generated by computer programs are not direct 'copies' or 'reproductions' of the literary or substantive content of the computer programs. This distinction results from the fact that the same screen can be created by a variety of separate and independent computer programs."); Midway Mfg. Co. v. Strohon, 564 F. Supp. 741 (N.D. Ill. 1983) ("It is quite possible to design a game that would infringe Midway's audiovisual copyright but would use an entirely different computer program.").

120. John Yates, a computer law specialist, commented that Lotus prevailing over Paperback Software "is going to re-emphasize that screen displays are copyrightable subject matter even if the underlying code is different in the two products." William M. Bulkeley, *Lotus Development Wins Suit Changing Paperback Software Violated Copyright*, WALL ST. J., June 29, 1990, at B4.

provides the comprehensive and flexible protection that graphical user interfaces demand.

III. TRADE DRESS FILLS THE VOID FOR SOFTWARE ELEMENTS UNDER PATENT AND COPYRIGHT LAWS

Although design patents may offer some, albeit limited, protection for graphical user interface elements, copyright law is not capable of being stretched that far.[121] Trade dress principles can be applied consistently to graphical user interface elements and provide several advantages.

The threshold inquiry under trade dress law, which is a subset of trademark law, is whether the product indicates a source of the product in the minds of the relevant consuming public.[122] Certainly, graphical user interfaces are capable of signifying the source of the product. Proceeding past this threshold question, several advantages are apparent when trade dress law is applied to graphical user interfaces, which includes a well developed and applicable test for infringement.[123] Also, the graphical display is considered as a whole, which is consistent with how the display is perceived by the consumer.[124] Finally, after successful graphical user interface elements are protected, competition in this important area of the software market is advanced because the stimulus to develop new and potentially better products is promoted.[125]

A. Trademark and Trade Dress Primer

The Lanham Act of 1946[126] is the statutory foundation for federal trademark law. Specifically, section 43(a)[127] is the primary vehicle for

121. Broderbund Software, Inc. v. Unison World, Inc., 648 F. Supp. 1127, 1133, 231 U.S.P.Q. (BNA) 700, 703 (N.D. Cal. 1986) (the court extended copyright protection to the screen display).

122. *See* S. Rep. No. 79-1333, *reprinted in* 1946 U.S.C.C.A.N. 1274-75; AmBrit, Inc. v. Kraft, Inc., 805 F.2d 974, 979, 1 U.S.P.Q.2d (BNA) 1161, 1164 (11th Cir. 1986).

123. *See infra* Part III.B.4 for discussion on infringement analysis under trade dress law.

124. Extracting individual elements from the graphical user interface is illogical in light of the manner in which the computer user views the software interface system.

125. *Cf.* David Bank, *Software Firms Look Outside Windows to Handle Data*, WALL ST. J., June 27, 1997, at B4 (competition provides computer manufacturers a stimulus to develop new and better programs).

126. 15 U.S.C. §§ 1051-1127 (1994).

127. Section 43 (a) provides:

(a) Civil action

the assertion of trade dress infringement in the federal courts, and Congress has recently acted to codify trade dress in the Lanham Act with the introduction of the "Trade Dress Protection Act"[128] on February 5, 1998.

Trademark law is premised on the principle that the proposed entity indicate source.[129] The product, or an identifiable feature of the product, needs to indicate to relevant consumers that the product's source is one entity.[130] Trademarks are significant because they enable users of the product to identify those goods or producers[131] with which they have

---

(1) Any person who, on or in connection with any goods or services, or any container for goods, uses in commerce any word, term, name, symbol, or device, or any combination thereof, or any false designation or origin, false or misleading description of fact . . . which

(A) is likely to cause confusion, or to cause mistake, or to deceive as to the affiliation, connection, or association of such person with another person, or as to the origin, sponsorship, or approval of his or her goods, services, or commercial activities by another person, or

(B) in commercial advertising or promotion, misrepresents the nature, characteristics, qualities, or geographic origin of his or her or another person's goods, services, or commercial activities,

shall be liable in a civil action by any person who believes that he or she is or is likely to be damaged by such act.

15 U.S.C. § 1125(a) (1994).

128. H.R. 3136, 105th Cong. (1998). The Trade Dress Protection Act was introduced by the Hon. Howard Coble of North Carolina on February 5, 1998. There are two provisions in the act relevant to software protection. First, inherently distinctive trade dress is protectible under federal trademark law without a showing that it has acquired secondary meaning. Second, the act provides the Patent and Trademark Office with a statutory basis upon which it may refuse to register matter that, as a whole, is functional as a whole.

129. 15 U.S.C. §1127 (1994) (defining a trademark as "any word, name, symbol or device or any combination thereof . . . used . . . to identify and distinguish his or her goods, including a unique product, from those manufactured or sold by others and to indicate the source of the goods, even if that source is unknown"); Craig Allen Nard, *Mainstreaming Trade Dress Law: The Rise and Fall of Secondary Meaning*, DET. C.L. REV. 37, 44-45 (1993) (The legislative history to the Lanham Act reveals three underlying policies. First, to protect the public from counterfeit marks and false trade description; second, to protect the goodwill that a trademark owner has built up; third to stimulate and encourage competition.).

130. For example, Owens-Corning has been successful in educating consumers that when consumers view pink insulation on a home under construction, there is one source for pink insulation. *In re* Owens-Corning Fiberglass Corp., 774 F.2d 1116, 227 U.S.P.Q. (BNA) 417 (Fed. Cir. 1985).

131. A relative continuum of trademarks exist: "KODAK" is an examples of a fanciful mark which is the strongest trademark. The mark has no meaning other than as a combination of letters used as a trademark. The words used as marks bear no relation to the products to which they are applied. Next are suggestive marks and are trademarks which require some thought as to what products the trademark represents. Descriptive words, the next category of trademarks, describe the items to which they are applied and are only capable of functioning as trademarks if they have acquired secondary meaning: a significant number of

good, or bad, experiences[132] and provide an indicia of consistent quality for consumers.[133] In addition, trademarks encourage the production of better products by fostering competition among producers.[134]

Trade dress is a subset of trademark law and is a judicial creation.[135] Both trademark law and trade dress law are statutorily protected under the Lanham Act and share underlying legislative policies.[136] The Supreme Court, in *Taco Cabana, Inc. v. Two Pesos Inc.*,[137] provides a significant basis upon which to argue that certain characteristics or combinations of elements that significantly represent a product in the minds of consumers are worthy of trademark protection.[138]

In order to obtain trade dress protection, a manufacturer must establish that: (1) the product configuration is either inherently distinctive or has acquired secondary meaning[139] in the marketplace, thus identify-

---

consumers identify a source for that product. The weakest marks are generic words and are not capable of functioning as trademarks. *See* Abercrombie & Fitch Co. v. Hunting World, Inc., 537 F.2d 4, 189 U.S.P.Q. 759 (2d Cir. 1976).

132. W.T. Rogers Co. v. Keene, 778 F.2d 334, 338, 228 U.S.P.Q. 145, 146 (7th Cir. 1985) (the purpose of "[trademark protection] is to reduce the cost of information to consumers by making it easy for them to identify the products or producers with which they have had either good experiences, so they want to keep buying the product, or bad experiences, so that they want to avoid the product or the producer in the future.").

133. S. 1883, 100th Cong., CONG. REC. H10411, H10419 (daily ed. Oct. 19, 1988) (remarks of Rep. Kastenmeier) ("Trademarks allow consumers to identify and intelligently pick and choose among products."); *see generally* Daniel J. Gifford, *The Interplay of Product Definition, Design and Trade Dress*, 75 MINN. L. REV. 769 (1991) (arguing that trademark law and the competition that is stimulated encourages firms to develop new products).

134. Landes & Posner, *supra* note 108, at 270.

135. *See* Two Pesos, Inc. v. Taco Cabana, Inc., 505 U.S. 763, 764-65 n.1, 23 U.S.P.Q.2d (BNA) 1081, 1082 n.1 (1992) (the Court states that "trade dress of a product is essentially its total image and overall appearance."); John H. Harland Co. v. Clarke Checks, Inc., 711 F.2d 966, 980, 219 U.S.P.Q. (BNA) 515, 528 (11th Cir. 1983) ("Trade dress involves the total image of a product and may include features such as size, shape, color or color combinations, texture, graphics, or even particular sales techniques.") (citations omitted).

136. The legislative history of the 1946 Lanham Act states: "There is no essential difference between trade-mark infringement and what is loosely called unfair competition. Unfair competition is the genus of which trade-mark infringement is one of the species .... All trade-mark cases are cases of unfair competition and involve the same legal wrong." H.R. REP. No. 944, 76th Cong., 1st Sess. § 3 (1939); Blau Plumbing, Inc. v. S.O.S. Fix-It, Inc., 781 F.2d 604, 608, 228 U.S.P.Q. (BNA) 519, 520 (7th Cir. 1986) (J. Posner stated that "there was probably no substantive legal difference between [trade dress and trademark].").

137. Two Pesos, Inc. v. Taco Cabana, Inc., 505 U.S. 763, 23 U.S.P.Q.2d (BNA) 1081 (1992).

138. *Id.* at 780, 23 U.S.P.Q. (BNA) at 1088.

139. *See* Nard, *supra* note, 128; *see also* Stormy Clime, Ltd. v. Progroup, Inc., 809 F.2d 971, 974, 1 U.S.P.Q.2d (BNA) 2026, 2028 (2d Cir. 1987); Ambrit, Inc. v. Kraft, Inc., 805 F.2d 974, 978 (11th Cir. 1986), *cert. denied*, 481 U.S. 1041 (1987).

ing the source of the good;[140] (2) the design as a whole[141] is nonfunctional;[142] and (3) the likelihood of confusion as to source or origin is created by the similarities between the plaintiff's and defendant's designs.[143]

Graphical user interface elements represent the computer software program to the computer user in addition to allowing the user to interact with the software program. The choices that a software producer makes in designing the program are unlimited, including the colors, shapes, configuration, and use of the features of the screen. Notwithstanding the similarities of the graphical user interface elements on current software programs, the screen displays created by the software program are capable of identifying the source of the program.[144]

B. How Trademark and Trade Dress Apply in the Context of Software

1. Trade Dress Allows the Graphical User Interface to be Viewed as a Whole

Proper analysis for the likelihood of confusion should occur "in their entirety and in context."[145] The look and feel of software, the graphical

---

140. Knitwaves, Inc. v. Lollytogs Ltd., 71 F.3d 996, 1004-07, 36 U.S.P.Q.2d (BNA) 1737, 1443-46 (2d Cir. 1995).

141. At least one commentator argues that computer icons, the miniature pictures that appear on the computer screen, are suitable for trademark protection. This is a difficult argument when the icons are viewed in isolation and not as part of the design of the screen itself. *See* Matt D. Carlson, Comment, *Intellectual Property Protection For Computer Icons: The Trademark Alternative*, 31 U.S.F. L. REV. 433 (1997).

142. Inwood Labs. v. Ives Labs., 456 U.S. 844, 850-51 n.10, 214 U.S.P.Q. (BNA) 1, 4 n.10 (1982); *see also* Qualitex Co. v. Jacobson Prods. Co., 514 U.S. 159, 165, 34 U.S.P.Q.2d (BNA) 1161, 1163 (1995).

143. Polaroid Corp. v. Polaroid Elecs. Corp., 287 F.2d 492,495, 128 U.S.P.Q. (BNA) 411, 413 (2d Cir. 1961). The Second Circuit's likelihood of confusion test uses eight factors:

> The strength of the mark, the degree of similarity between the two marks, the proximity of the products, the likelihood that the prior owner will bridge the gap, actual confusion, and the reciprocal of defendants good faith in adopting its own mark, the quality of defendant's product, and the sophistication of the buyers.

Id.

144. Protection efforts for the original graphical user interface systems have failed and therefore most operating systems look alike. Consequently, users may confuse one software manufacturer with another. As software, and the graphical user interface elements in particular, develops and receives adequate protection under the law, the differences and source denotations will become more pronounced.

145. Levi Strauss & Co. v. Blue Bell, Inc., 632 F.2d 817, 822, 208 U.S.P.Q. (BNA) 713, 718 (9th Cir. 1980) (the overall impression of the mark and not the individual features matter when evaluating registrability of a trademark).

user interface, must be viewed in its entirety because that is how the computer user encounters the software program.[146] As discussed previously, the extraction of individual elements from the graphical user interface renders the analysis inappropriate for products such as software.

2. Graphic User Interfaces Indicate Source

Most user interfaces create a distinct impression on the user, and certainly graphical user interfaces are capable of indicating source when the graphics are used by the potential trademark holder as a source indicating feature. Because the original developers of the Xerox Star User Interface failed to protect their original product and subsequent competitors took advantage of that fact, many software programs on the market today look very much alike.[147] Had Xerox protected their important user interface innovation, competitors would have been forced to develop their own distinctive products.

3. Graphic User Interfaces as a Whole are Non-functional

The fundamental policy goal of the functionality doctrine in trademark law is to balance monopoly rights with public access.[148] The tension relates to the fear that an inventor may extend a monopoly over a product's functional features through the use of trademark law[149] when a patent would expire after a specified period.[150]

Trade dress and product configuration will not be accorded protection under trademark law principles if the trade dress or product configuration "contributes to the efficiency or economy in manufacturing or handling, or to durability."[151] Courts have had some difficulty dis-

146. Safeway Stores, Inc. v. Safeway Discount Drugs, Inc., 675 F.2d 1160, 1165, 216 U.S.P.Q. (BNA) 599, 603 (11th Cir. 1982).

147. *See* Apple Computer, Inc. v. Microsoft Corp., 759 F. Supp. 1444, 1454, 18 U.S.P.Q. 2d (BNA) 1097, 1103-04 (N.D. Cal. 1991); *see also supra* note 26.

148. Tom W. Bell, *Virtual Trade Dress: A Very Real Problem,* 56 MD. L. REV. 384, 416 (1997); Michael Garvey, *Product Configuration TM: Shape of Things to Come?,* (visited Feb. 26,1998) <http://www.ljx.com/trademark/tmcomf.html>.

149. Qualitex Co. v. Jacobson Prods. Co., 514 U.S. 159, 164-65, 34 U.S.P.Q.2d (BNA) 1161 (1995), (the Court stated: "If a product's functional features could be used as trademarks . . . a monopoly over such features could be obtained without regard to whether they qualify a patents and could be extended forever . . . .").

150. *See infra* Part I(B)(2); *see also* Kenneth L. Port, *The Illegitimacy of Trademark Incontestability,* 26 IND. L. REV. 519, 523-527 (1993) (arguing that the ability to register a trademark as an incontestable trademark allows the holder of the trademark to have too many rights).

151. 1 MCCARTHY, *supra* note 84, § 8:20, 8-48-49; *see also* Inwood Lab. v. Ives Labs.,

cerning whether to apply this test narrowly or more broadly when attempting to categorize a particular product in relation to competitors' products.[152] The distinction becomes important when faced with a product such as a software program.

The graphical user interface is perceived by the user in its entirety. Therefore, the functionality inquiry should focus on the totality of the display and not on individual elements as discrete features.[153] Recent court decisions have made clear that the functional nature of *some* of the product features does not preclude protection because the trade dress as a whole is how the user views the product.[154]

It is illustrative to consider the "roots" of computer software. Before graphical user interfaces were developed in the early 1970s, it was necessary to type in the name of the computer program at the flashing "prompt." Certainly, this manner of user interface would not be capable of functioning as a source indicator. Currently, however, most word processing programs look alike, and commentators argue that the utilitarian aspects preclude the user interface from being protected under trademark laws. This argument is circular in that the popularity of the desktop metaphor makes it widely accepted and it therefore is functional. If and when a software producer develops and markets a completely new user interface system, perhaps the use of a "cockpit metaphor" or other creative idea, the desktop metaphor would appropriately be viewed as one of many configurations for a word processing program or operating system.

4. Infringement Analysis is Applicable

Infringement under copyright is sorely inadequate. This point is illustrated when a judge attempts to determine if infringement existed by examining the "pixels" in the software programs in relation to one an-

---

456 U.S. 844, 851 n.10, 214 U.S.P.Q. (BNA) 1, 4 n.10 (1982).

152. *See, e.g.*, Topps Co. v. Gerrit J. Verburg Co., 41 U.S.P.Q.2d (BNA) 1412, 1419; (S.D.N.Y. 1996) (court construed product line narrowly, determining that product line is "diamond engagement shaped lollipops" as opposed to a more broad category such as "candy").

153. Marketing Displays, Inc. v. Traffix Devices, Inc., 967 F. Supp. 953, 960, 43 U.S.P.Q.2d (BNA) 1865, 1874 (E.D. Mich. 1997).

154. *See, e.g.*, Sengoku Works Ltd. v. RMC Int'l, Ltd., 96 F.3d 1217, 40 U.S.P.Q.2d (BNA) 1149 (9th Cir. 1996) (holding that the controls of kerosene heaters as a whole are nonfunctional); Marketing Displays, Inc. v. Traffix Devices, Inc., 971 F. Supp. 262, 272, 43 U.S.P.Q.2d (BNA) 1865, 1874-75 (E.D. Mich. 1997) (stating that product configuration must be viewed in its an entirety); Pebble Beach Co. v. Tour 18 Ltd., 942 F. Supp. 1513 (S.D. Tex. 1996) (finding golf course design nonfunctional).

other and set forth in the court opinion.[155] Surely a system of protection can be devised that recognizes that important elements are infringed even though the dialogue box outline is of differing widths.[156]

Under trade dress, infringement analysis would be consistent with that employed for trademarks.[157] A court would examine the marks and determine the likelihood of confusion[158] and relevant factors that would be considered include:

1. The strength of the plaintiff's mark;
2. Similarity of the plaintiff's and defendant's marks;
3. Proximity of the feature in relation to the defendant's;
4. Evidence of actual confusion;
5. Good faith of the defendant in adopting the trademark; [and]
6. Sophistication of the consumer and the nature of the trade mark.

The focus of a court's inquiry is on the consumer who will be presented with software programs and their features in context. For example, consider the facts in *Lotus Development Corp. v. Paperback Software.*[159] If one examines Borland's copying of the Lotus 1-2-3 software in light of these factors, the result would likely by consistent with the District Court of Massachusetts: Infringement (copying) existed and Borland took too much of the Lotus 1-2-3 software.

---

155. Apple Computer, Inc. v. Microsoft Corp., 821 F. Supp. 616, 619-20, 27 U.S.P.Q.2d (BNA) 1081, 1085-86, (N.D. Cal. 1993).

156. *Id.* at 620-21, 27 U.S.P.Q.2d (BNA) at 1086-87.

157. 15 U.S.C. § 1114.

158. Each federal circuit has developed a comparable test using these factors to determine the likeliness of confusion. *See* Imagineering, Inc. v. Van Klassens, Inc., 53 F.3d 1260, 1264, 34 U.S.P.Q.2d (BNA) 1526, 1529 (Fed. Cir. 1995); Fissons Horticulture, Inc. v. Vigoro Indus., 30 F.3d 466, 473, 31 U.S.P.Q.2d (BNA) 1592, 1597 (3d Cir. 1994); Babbit Elecs., Inc. v. Dynascan Corp., 38 F.3d 1161, 1178, 33 U.S.P.Q.(BNA) 2d 1001, 1013 (11th Cir. 1994); Smith Fiberglass Prods., Inc. v. Ameron, Inc., 7 F.3d 1327, 1329, 28 U.S.P.Q.2d (BNA) 1614, 1616 (7th Cir. 1993); DeCosta v. Viacom Int'l, Inc., 981 F.2d 602, 606, 25 U.S.P.Q.2d (BNA) 1187,1190 (1st Cir. 1992); Blue Bell Bio-Medical v. Cin-Bad, Inc., 864 F.2d 1253,1260, 9 U.S.P.Q.2d (BNA) 1870,1873 (5th Cir. 1989); Jordache Enters. V. Hogg Wyld, Ltd., 828 F.2d 1482, 1484, 4 U.S.P.Q.2d (BNA) 1216, 1218 (10th Cir. 1987); Co-Rect Prods., Inc. v. Marvy! Adver. Photography, Inc., 780 F.2d 1324, 1330, 228 U.S.P.Q. (BNA) 429, 432 (8th Cir. 1985); Pizzeria Uno Corp. v. Temple, 747 F.2d 1522, 1527, 224 U.S.P.Q. (BNA) 185, 187 (4th Cir. 1984); Frisch's Restaurants, Inc. v. Elby's Big Boy of Steubenville, Inc., 670 F.2d 642, 648, 214 U.S.P.Q. (BNA) 15, 19 (6th Cir. 1982); AMF Inc. v. Sleekcraft Boats, 599 F.2d 341, 348-49, 204 U.S.P.Q. (BNA) 808, 814 (9th Cir. 1979); Polaroid Corp. v. Polarad Elecs. Corp., 287 F.2d 492, 495, 128 U.S.P.Q. (BNA) 411, 413 (2d Cir. 1961).

159. 740 F. Supp. 37, 15 U.S.P.Q.2d (BNA) 1577 (Mass. Dist. Ct. 1990).

The Lotus 1-2-3 software, to begin with, has strong recognition from the consumers who use that product. The menu designs and the command structure have been features of their program that greatly impact the computer as a whole and the manner in which consumers use the program. Because Lotus has developed and marketed these products and have "taught" the public about these product features, they have become inherently distinctive and an indicator of the source of the product. Further, when the factual similarities of *Two Pesos, Inc. v. Taco Cabana, Inc.*[160] and *Lotus Development Corp. v. Paperback Software, Inc.*[161] are analyzed, it is difficult to conclude that based upon the inherent qualities of the products at issue that user interface elements should not be protected.

The second factor, the similarity between the plaintiff's and defendant's marks, would cut strongly against Borland. The resulting products, after copying, were virtually identical. Borland made no secret of that fact.

The intention of Borland to design a product that appeared to be their competitor's product evidences the lack of good faith on Borland's part. The company actively sought to copy Lotus' menu structures and commands. This factor cuts against Borland. Other factors such as the quality of the products and the sophistication of the buyer would also be factors in Lotus Development Company's favor.

The criteria for infringement is predictable and concrete. Under copyright analysis, it is unclear whether a program feature is an idea or an expression of an idea, and whether the copied computer code is substantially similar to another product. These difficult distinctions are largely unnecessary under a trademark analysis.

C. The Benefits of the Application Of Trade Dress

1. Innovative, Not Duplication, Would be the Norm

The current ability of software programmers to substantially use their competitor's graphical user interface elements stifles the growth of new programs. Any computer user need only walk around the office complex to determine that pull down menus, mice, and icons have become the norm for personal desktop computing.

160. 505 U.S. 763, 23 U.S.P.Q.2d (BNA) 1081 (1992).

161. 740 F. Supp. 37, 15 U.S.P.Q.2d (BNA) 1577 (Mass. Dist. Ct. 1990).

Since April of 1981, when Xerox brought their new computer designed for offices to the market, products in the personal computer user interface market have been repetitious.[162] The present circumstances indicate that the major players in the software industry have been able to get by with the current user interface model instead of injecting research dollars and human resources into producing a better computer interface.[163] Because the legal protection for product development is, at best, questionable with regard to computer user interface elements, capital intensive projects devoted to developing new graphical user interfaces are unlikely to receive budgetary allocations. At worst, their potentially pioneering effort would be passed around to competitors because the shortcomings of copyright and patent laws to offer adequate protection.

At some point when an adequate protection scheme is in place, the words of one journalist may ring true: "The new frontier is beyond Windows."[164]

2. Intent-to-Use System Offers Market Protection

Authors of the *A Manifesto Concerning the Legal Protection of Computer Programs* ("*Manifesto*"),[165] argue that the largest flaw in copyright protection is that competitors are able to legally access and copy the computer program before the developer has the ability to realize economic gains.[166] Because the costs of research and development can be quickly subverted by a competitor's reverse engineering, the possibility of failing to recover these costs is very real in the software market.[167]

When trademark laws are applied, this seemingly large obstacle is addressed by the availability of the intent to use system.[168] When a producer plans to use a product in commerce, the intent to use registration system provides protection from the chilling effect caused by uncer-

---

162. Smith, *supra* note 53, at 242.

163. "A hardy band of new software companies is seeking to change the very way people look at information.... Someday, they assert, visual schemes may replace Microsoft Corp.'s best-selling Windows operating system as a fundamental way to organize information." David Bank, *Software Firms Look Outside Windows to Handle Data: Visual Navigation Techniques May Help Alleviate Information Overload*, WALL ST. J., June 27, 1997, at B4.

164. *Id.*

165. *See Manifesto*, *supra* note 7.

166. *Id.* at 2429-30.

167. *Id.*

168. 15 U.S.C. § 1051(b) (1994).

tainty. The producer also has access to reliable information as to the availability of a trademarked feature.[169] Further, the three years and subsequent extensions allowed under the intent to use provision allows the producer to continue to develop the software in preparation of marketing knowing that the expenditures toward that end will not be subverted in the marketplace. Under such a system, business planning is facilitated.

In order to properly register a product, it would be necessary for the trade dress to be inherently distinctive. Circuits are split on the issue of how this showing occurs;[170] however, the proposed amendment to section 2(g) of the Lanham Act would facilitate protection of inherently distinctive trade dress.[171] If passed, Congress would codify the holding in *Taco Cabana* that a product is inherently distinctive if the relevant public is likely to identify the source of the product.[172] The focus of the inquiry would be the impact of the relevant public.[173]

An additional benefit of Intent-to-Use provisions is that it allows smaller software developers active in peripheral markets to compete with the multi-billion dollar software producers. When an Intent-to-Use application is filed for a development that is the production stage, all computer developers receive notice and may begin to develop additional products. In this way, additional markets are created.

Intent-to-Use provisions of the Lanham Act operate to effectively protect the research and development costs of software developers, as well as provide sound economic incentives to bring new products to the market. Graphical user interface elements are capable of indicating the source of the product provided innovative products are created.

---

169. S. REP. NO. 100-515, 4-6, 22-26, 29-32, 38 (1988).

170. *See, e.g.*, Stuart Hall Co. v. Ampad Corp., 51 F.3d 780, 785, 34 U.S.P.Q.2d (BNA) 1428, 1431 (8th Cir. 1995) (stating marks that are arbitrary, fanciful or suggestive are inherently distinctive); Knitwaves, Inc. v. Lollytogs Ltd., 71 F. 3d 996, 1008, 36 U.S.P.Q.2d (BNA) 1737, 1746-47 (2d Cir. 1995) (showing that it is necessary to prove that the primary purpose behind the design was to identify the product's source); Duraco Prods. v. Joy Plastic Enters., 40 F.3d 1431, 1432, 32 U.S.P.Q. (BNA) 1724, 1725 (3d Cir. 1994) (stating that it is necessary to provide proof that trade dress is "(i) unusual and memorable, (ii) conceptually separable from the product, (iii) and likely to serve as a designator of origin of the product.").

171. 144 CONG. REC. E117-02 (daily ed. Feb. 5, 1998) (statement of Hon. Howard Coble).

172. *Taco Cabana*, 505 U.S. at 776, 23 U.S.P.Q.2d (BNA) at 1086 (1992) (holding that trade dress that is inherently distinctive is protectible without a showing of secondary meaning).

173. 144 CONG. REC. E117-02 (daily ed. Feb. 5, 1998) (statement of Hon. Howard Coble).

3. The Danger Of Enacting Sweeping And Outdated Reforms to the Law is Avoided

One commentator outlines the possible dangers of enacting new and comprehensive legislation for dealing with the onslaught of computer technology.[174] Her points are well taken. Because the legislative process is time-consuming and laborious, the changes would very likely be outdated before they are enacted.

By using existing trademark law for elements of the program that are suited for trademark protection in conjunction with copyright and patent laws, the statutory schemes and case law is suitably developed to provide a guidelines and a framework within the industry and in academia.

4. Standardization Can Be Achieved

Standardization in the world of computer software involves two concepts. One aspect of standardization is producing software programs that appear familiar to users enabling them to move between products easily.[175] The second aspect of standardization is involves a secondary market related to both hardware and software products consisting of add-on or accessory products.

Protecting graphical user interfaces through trade dress accomplishes favorable results for both objectives. The first type, user familiarization across product lines, should not inhibit the protection of innovative software interface elements.[176] Instead, corporations should be rewarded through a strong protection system for their innovations through trade dress protection.[177] Strong protection measures encourage innovation and provides the market with competition. Consequently, technology will improve on existing graphical user interface elements, software programs will become more intuitive, and training costs will correspondingly decrease.

---

174. Ginsburg, *supra* note 9, at 2562-63.

175. See Gerard J. Lewis Jr., *Lotus Development Corp. v. Paperback Software International: Broad Copyright Protection For User Interfaces Ignores The Software Industry's Trend Toward Standardization*, 52 U. PITT. L. REV. 689 (1991) (arguing for narrow protection of graphical user interfaces thereby increasing similarity across producers and benefiting consumers).

176. *But see* Lewis, *supra* note 160; Timothy S. Teter, *Merger and the Machines: An Analysis of the Pro-Compatibility Trend in Computer Software Copyright Cases*, 45 STAN. L. REV. 1061 (1993).

177. *See* Landes & Posner, *supra* note 108, at 270-75.

The second type of standardization, which most often encourages smaller businesses to produce products on the secondary market, would actually benefit through trade dress protection of graphical user interfaces. Called peripheral products, users benefit from being able to select a variety of compatible products.[178]

The capability of product lines to compatibly interact between producers will benefit consumers because it allows a variety of choices as well as differences among product lines.[179] The ability of software to be compatible in enhanced when producers have access to the underlying code, or literal elements, of the software. Trade dress protection offers the benefit of protecting the non-literal graphical user interface because the literal elements of the program remain in the public domain. Therefore, the peripheral product market is enhanced.[180]

Affording no protection, or inconsistent protection, operates as a disincentive for software producers and stalls innovation and development in the market as a whole.[181] When source indicative and technically progressive products are created, it is counterproductive to allow such innovations to be unprotected for the sake of standardization. When left to a free market, compatibility may occur on its own, or consumers will simply compete for market share.[182]

CONCLUSION

178. International Business Machines dramatically reshaped the computer industry in 1964 through the well publicized launching of the IBM System/360. The hallmark of this product line was the range of software compatible products. MARTIN CAMPBELL-KELLY & WILLIAM ASPRAY, COMPUTER: A HISTORY OF THE INFORMATION MACHINE 257 (1996).

179. For example, Microsoft produces Word, Excel, and Powerpoint. Each are available separately or can be purchased as a package. The software programs interact easily and offer the user some advantages when creating large products.

180. When the technology is in the public domain, it is available for researchers and developers to use and to make improvements. The computer hardware industry illustrates this point when in the 1980s most of the popular software packages were converted to run on the IBM personal computer which became the industry standard. The popularity of both the software and the hardware encouraged other manufacturers to produce "clone" machines which standardized the personal computer market. It was easy to produce the clone machines because the Intel 8088 microprocessor was available on the open market. CAMPBELL-KELLY & ASPREY, *supra* note 178, at 257.

181. Landes & Posner, *supra* note 108, at 270.

182. *See, e.g.*, Dean Takahashi, *Microsoft, Netscape Agree On Standard for 3-D on Internet*, WALL ST. J., Aug. 4, 1997, at B8 ("Microsoft and Netscape Communications Corp. have agreed to popularize a single standard for viewing three-dimensional images on the Internet."); Nate Zelnick, *Microsoft To Develop for Unix, Support Attempts to Bridge CORBA, COM Standards*, INTERNET WORLD, Feb. 2, 1998, at 4.

The confusion that currently exists related to protection of graphical user interfaces is directly related to the failure of Xerox to protect the first graphical user interface elements. Apple popularized the desktop metaphor that Xerox created.[183] Subsequent competitors, most notably Microsoft, quickly realized the magnitude of the graphical user interface elements and proceeded to mimic them or to seek a license.[184]

The result is that graphical user interfaces have really not progressed all that much since Xerox's pioneering effort. In fact, the desktop metaphor has become the standard and judges actually believe that it is so basic to software that it is called an idea and not an expression of an idea.

It is not too late. Future innovations should be rewarded with protection. And now is the time for software manufacturers to seek trade dress protection for graphical user interface features that indicate the source of the product. Graphical user interfaces, those portions that "are" the computer to users, indicate source and warrant protection under trade dress law.

JANE M. ROLLING

183. Curtis, *supra* note 5, at 52. (noting that Apple introduced the Macintosh during half-time of the 1984 Super Bowl).

184. Using the term Mimic is being kind . . . the "trash can" was turned into a politically correct "recycle bin." The concept itself was copied however. *See* Apple Computer, Inc. v. Microsoft Corp., 821 F. Supp. 616, 619-20, 27 U.S.P.Q.2d (BNA) 1081, 1085-86, (N.D. Cal. 1993).