
Sam Kraemer
DISCLOSURE AND THE COMPUTER PROGRAMMER

Americans today are constantly exposed to the wonders and problems of data processing. Because of the technology's youth, however, most of us are ill-prepared to work closely with computers. When the need to use one arises, most people hire industry-trained specialists to oversee their operations. In the business world, systems analysts and programmers are hired to create computer systems that will perform specific functions. Since computers often play a significant role in a business' prosperity, their failure to function properly can mean business failure. As a result, the computer programmer is often placed in a powerful position: the client needs the programmer's services in order to stay in business.

This Note considers the law's ability to protect the client from being taken advantage of by the programmer. It first addresses the role of contract law, considering whether the programmer may have obligations beyond those expressly set forth in the contract. It considers whether the programmer's technical knowledge and the client's reliance on the programmer are sufficient to create a more stringent fiduciary duty on the part of the programmer.

The Note then looks at tort law to determine the standard of care owed the client by the programmer/analyst. It is arguable that because of his or her expertise, the programmer/analyst should be held to a higher standard of care than the normal "average reasonable person" standard. The law presently has stricter tort standards for certain learned occupations. This Note considers whether the policy considerations behind the imposition of these higher standards apply to computer programmers and analysts.

At present, the computer-ignorant businessperson has few legal tools available to keep a programmer from exploiting his or her ignorance. While this situation is understandable given the computer field's infancy, it is time the law addressed this problem created by the new technology.

I. THE BUSINESSPERSON'S PROBLEM

Computers perform hundreds of business services, from keeping inventory records to producing financial statements. Often the task they
perform is essential to the business' success.¹ The buyer must therefore know exactly what he is getting when he contracts for a computer program. The need to understand programming technicalities creates inequalities in bargaining power between the computer professional and the client. While the businessperson is generally interested only in the ability of the computer system to achieve the expected results, the programmer's concerns lie in the internal process used in creating the system. The programmer/analyst develops that process by writing a program composed of source code statements. An equally important part of a complete program, however, is the documentation—something of which the average businessperson is unaware. Documentation consists of normal English sentences in paragraph form setting forth the step-by-step operation of the program. Documentation allows any programmer to understand the program and to correct errors within it. Because the average businessperson is unaware of the need for documentation, he or she may assent to a software contract that fails to require the programmer to provide documentation. Without it, an error in the system can be fixed only by the original author of the program. The businessperson is thus at the original programmer's mercy: without the services of this particular programmer the system, and potentially the business, will fail. The law should address the vulnerability of the businessperson in this relationship.

To decide what legal solution would best remedy this situation, the ideal relationship between the programmer and the client must first be determined. The underlying problem is the discrepancy in computer knowledge between the two parties. The analyst knows the intricacies of the program; the average businessperson does not. If the disparity in knowledge were eliminated, then neither of the parties would have this advantage. Thus any solution to the problem should involve education of the client. The client must understand the function and necessity of documentation as well as source code. Disclosure of this information would enable the client to make an informed decision when purchasing computer software, and would eliminate this opportunity for the programmer to take advantage of the client.

The programmer is in the best position to provide this information to the client, because he or she has the necessary information. Requiring the programmer to educate the client in this way during negotiations would impose only a slight burden on the programmer. Alternatively, if the client were made responsible for obtaining the necessary information, the burden imposed would be much greater. Both the time and money that would be expended by the businessperson locating and consulting with a computer expert would be far greater than

the expense to the programmer of disclosing the appropriate information. Thus imposing a duty of disclosure on the programmer would best solve the problem of unequal information.

Such a duty of disclosure in the programmer should be imposed through legislation. A law requiring every program to be accompanied by documentation, however, would be economically impractical since the client would ultimatley bear the cost. Such legislation would deprive the businessperson of the opportunity to decide in any given software contract whether the cost of the documentation was justified by the risk of not having it. A more practical statute would simply require the programmer to make the necessary disclosure to the client. Such a statute would solve the problem of unequal knowledge by placing the burden of equalizing knowledge on the programmer. Since no legislature seems ready to take this step, other means of reducing the businessperson's vulnerability must be found in the mean time.

Contract law may provide the businessperson several theories to require the programmer to reveal the necessary information. First, some contractual relationships give rise to fiduciary duties.\textsuperscript{2} Courts have found a fiduciary duty where one contracting party has acquired domination over the other contracting party through the latter’s trust.\textsuperscript{3} A second theory under contract law that might be applicable is quasi-contract. To avoid injustice, a court will sometimes examine all of the aspects of a contract and modify it as equity demands.\textsuperscript{4} These two contractual theories—fiduciary duty and quasi-contract—may be useful in creating a balanced relationship between the programmer and the client.

Tort law provides another potential means of dealing with the programmer's ability to take advantage of the client's lack of knowledge. Every person's actions must meet a minimum legal standard of care. The normal standard of care is that of a reasonably prudent person under similar circumstances.\textsuperscript{5} Among the factors taken into account in determining the required standard of care is special knowledge possessed by the actor.\textsuperscript{6} Thus in judging the actions of a programmer/analyst, it must be determined whether he or she acted with the care of a reasonably prudent computer programmer. Case law does not require that the program be error-proof. To expect infallibility would be unreasonable: experts are expected only to act with reasonable and ordinary

\textsuperscript{2} See Hendrickson v. California Talc Co., 58 C.A. 467, 103 P.2d 806 (1942).
\textsuperscript{3} Id.
\textsuperscript{4} See 66 AM. JUR. 2D Restitution & Implied Contracts § 166 (1973).
skill.\textsuperscript{7} It is still unclear, however, whether this standard would prevent the programmer from taking advantage of the client. While the reasonably prudent person standard suffices in most situations, in certain contexts the courts have demanded more.\textsuperscript{8} If the programmer-client relationship falls within one of these exceptions, a professional standard of care might require the programmer to make the necessary disclosures. Each of the three legal theories examined—contract, quasi-contract, and tort—might be the basis for eliminating the bargaining advantages of the programmer. Each theory must be examined more closely to determine which would be most effective in remedying the situation.

II. CONTRACTUAL THEORIES

A. FIDUCIARY DUTY

A common way of demanding extra duties from a contracting party is to establish the existence of a fiduciary relationship.\textsuperscript{9} If fiduciary responsibilities are required of a person, that person is held to a much higher standard of care. As Justice Cardozo said in \textit{Meinhard v. Salmon},\textsuperscript{10} it is “not honesty alone, but the punctilio of an honor the most sensitive” that is then the standard of behavior. A fiduciary relationship arises “whenever confidence is reposed in one side, and domination and influence result on the other.”\textsuperscript{11} It exists where one party relies upon the judgment and advice of the other.\textsuperscript{12} “Fiduciary” is defined as “a person having a duty, created by his undertaking, to act primarily for another’s benefit in matters connected with such undertaking.”\textsuperscript{13} Out of this relationship “the law raises the rule that neither party may exert influence or pressure upon the other, take selfish advantage of his trust, or deal with the subject-matter of the trust in such a way as to benefit himself or prejudice the other.”\textsuperscript{14}

There are various theories about when a fiduciary relationship exists. For example, the commercial utility theory is used when a court feels it necessary to hold a person or a certain class of persons to a higher-than-average standard of ethics or good faith, in the interest of

\begin{itemize}
\item \textsuperscript{7} Gagne v. Bertran, 43 Cal. 2d 481, 489, 275 P.2d 15, 21 (1954).
\item \textsuperscript{8} In both the legal and medical professions, courts have held professionals to a higher standard of care. See, e.g., Canterbury v. Spence, 464 F.2d 772 (D.C. Cir. 1972), cert. denied, 409 U.S. 1064 (1984).
\item \textsuperscript{9} Bohlinger v. Word & Co., 34 N.J. Super. 583, 584, 113 A.3d 38, 42 (1955).
\item \textsuperscript{10} 249 N.Y. 458, 458, 164 N.E. 545, 546 (1928).
\item \textsuperscript{11} \textit{BLACK'S LAW DICTIONARY} 564 (5th ed. 1979).
\item \textsuperscript{12} Williams v. Griffin, 192 N.W.2d 283, 285 (Mich. 1971).
\item \textsuperscript{13} \textit{BLACK'S LAW DICTIONARY}, supra note 11, at 563.
\item \textsuperscript{14} Bohlinger v. Word & Co., 34 N.J. Super. 583, 584, 113 A.3d 38, 42 (1955).
\end{itemize}
protecting the integrity of a commercial enterprise. This theory assumes that trust is required in business dealings to achieve efficiency, and that only a higher standard of care will cause a businessperson to act in a trustworthy manner. A fiduciary relationship has also been considered contractual: the fiduciary is bound to protect or advance the interests of another because of the relationship between the parties. These theories of fiduciary relations demand an extra duty of care when one party to a contract may dominate the other.

Clearly a businessperson contracting with a programmer possesses less technical knowledge than the programmer. The programmer's knowledge alone, however, does not create a fiduciary duty. As discussed earlier, there must be a reposing of confidence and trust by the businessperson that results in domination by the programmer. It is arguable that by hiring a programmer/analyst, the businessperson is placing trust in him or her. The contract calls for a program, and the client trusts that everything will be included when the program is received. Therefore the programmer could be considered to have complete domination and influence over performance of the contract. Once the contract is signed, the only performance required of the client is cooperation with the programmer and payment of the fee. The businessperson informs the programmer of the desired function of the program and then waits for the finished product. Thus there is a placing of reliance by one party (the client) upon the judgment and advice of the other (the programmer). The programmer has complete control of the creation of the program, the objective of which is to benefit the client.

In addition to suggesting that the programmer-client relationship fits within the definition of a fiduciary relationship, it may also be argued that the commercial utility theory applies. A layperson using a computer is required to place the utmost trust in the programmer/analyst. The future of the client's business is being placed in the hands of the programmer, and the client must believe that the programmer is acting for the client's benefit. Therefore holding members of the computer profession to a higher standard of behavior might serve efficiency concerns. In sum, theories of fiduciary duty may well be applicable to computer programmers.

B. QUASI-CONTRACT

In addition to the express duties placed upon each party by the

---

16. Id. at 57.
17. Id. at 65.
18. See supra note 11 and accompanying text.
words of a contract, courts will sometimes look to equity and the implied requirement of "good faith and fair dealing" to require additional obligations from the parties involved. This requirement of "good faith" is discussed in numerous insurance company cases. In *Gruenberg v. Aetna Ins. Co.*\(^1\) the court stated that "there is an implied covenant of good faith and fair dealing in every contract that neither party will do anything which will injure the right of the other to receive the benefits of the agreement."\(^2\) This obligation to act fairly and in good faith when performing a contract is implied by law; it is not expressed in the contract itself.\(^3\) Breach of this duty may give rise to causes of action in both contract and tort.\(^4\) In the insurance setting, this means that the insurer has a duty to accept reasonable settlements\(^5\) and not to unreasonably withhold payments.\(^6\) Since good faith and fair dealing are required in the performance of every contract, an examination of what fairness requires may disclose whether implied duties exist in the computer programming context.

An implied duty of good-faith bargaining is needed in the programmer-client context. According to the court in *Gruenberg*, "neither party will do anything which will injure the right of the other party to receive the benefits of the agreement."\(^7\) In the programming context, the benefit of the contract to the client is the continued use of a working computer program. A program written without documentation provides the client with the benefit of the contract only so long as it functions properly. Computer errors are common, and without documentation repairing them is extremely difficult. Without documentation, only the original author can correct the errors. This situation clearly diminishes the benefits received by the businessperson, who is now at the mercy of one programmer for the use of the program. The program has been purchased outright, and the programmer should not possess this type of "lien" on the product. Any subsequent conflict between the two parties could lead to the business' downfall. The contract should be interpreted as calling not only for a program that works originally, but also for one that can be fixed if problems arise. By failing to inform the client of the need for documentation, the programmer hampers the businessperson's use of the program. Just as the duty of good faith requires an insurance company to accept reasonable settlements, it might be used to require programmers to disclose to clients the need for documentation.

---

2.  *Id.* at 573, 108 Cal. Rptr. at 484, 510 P.2d at 1038.
3.  *Id.* at 574, 108 Cal. Rptr. at 485, 510 P.2d at 1038.
4.  *Id.* at 573, 108 Cal. Rptr. at 484, 510 P.2d at 1038.
7.  *Id.* at 577, 108 Cal. Rptr. at 488, 510 P.2d at 1040.
This quasi-contract solution alone might be adequate to end domination by the programmer.

III. TORT THEORIES: A STANDARD OF CARE

No matter what conduct a person participates in, that person must act with a certain standard of care so as not to injure the rights of others. The normal standard of care is that of the average reasonable person. In certain circumstances, however, the law has decided that this normal standard is insufficient, and that a person must act with more care. Certain duties have been defined, and failure to comply with these duties constitutes a tort. The issue here is whether the average reasonable person standard is an adequate standard for programmers—that is, whether it sufficiently protects the client. If not, the programmer may fit the criteria that have been established for holding a person to a higher standard of care. Such a higher standard could potentially require disclosure by the programmer. An examination of tort law and “professional” standards of care is necessary to address these issues.

A. WHO HAS A HIGHER STANDARD OF CARE

The learned professions are typically held to a higher than normal standard of care. Physicians, lawyers, engineers, and accountants have all been sued under theories of professional negligence or malpractice. In *Lindner v. Barlow, Davis and Wood*, the court stated that “[a]ccountants have been recognized as a ‘skilled professional class . . . subject generally to the same rules of liability for negligence in the practice of their profession as are members of other skilled professions.’” The court seemed to suggest that “skilled professions” are occupations upon which the law will impose greater duties of care. If the programmer/analyst is considered a “skilled professional,” then a higher standard of care will be required.

B. WHO IS CONSIDERED A PROFESSIONAL

Defining “professional” is not simple. *Webster’s New Twentieth*
Century Dictionary defines "profession" as "a vocation or occupation requiring advanced training in some liberal art or science, and usually involving mental rather than manual work, as teaching, engineering, writing, etc.; especially medicine law, or theology." According to Dean Pound, a profession is "a group of men pursuing a learned art . . . in the spirit of public service." The Restatement of Torts states that a professional has "that special form of competence which is not part of the ordinary equipment of the reasonable man, but which is the result of acquired learning, and aptitude developed by special training and experience." While all of these definitions are helpful, they provide neither a clear nor complete understanding of the term. A closer examination of how an occupation becomes a profession is necessary to determine whether a programmer/analyst should be considered a professional.

The major characteristics of a profession seem to be included in the definitions above. The definitions, however, must be expanded upon. One commentator has suggested that in defining a professional, one must stress such criteria as advanced and specialized training along with continued use of intellect, discretion and judgment. The professional is seen as exercising intellectual judgment continually. Being a professional lends authority to one's statements, and enhanced status and reputation are determinant of earning capacity. Thus the status of "professional" is not determined solely by special training; every occupation has to some extent its own unique training. Rather, the question is whether the occupation requires continued intellectual judgment.

Some commentators claim that this listing of attributes is not the correct way to arrive at a valuable definition of a professional. Wilbert E. Moore, in his study The Professions: Roles and Rules, claims that "professionalism should properly be regarded as a scale rather than a cluster of attributes." In short, Moore defines a professional as one who practices a full-time occupation to earn a living. Many times, there is a commitment to a calling. The professional possesses special skills that come from training of exceptional duration and perhaps difficulty. The professional is service-oriented, attending to the needs of a client while using judgment and authority in advancing those needs. Fi-

34. WEBSTER'S NEW TWENTIETH CENTURY DICTIONARY 1437 (2d ed. 1978).
36. RESTATEMENT (SECOND) OF TORTS § 299A comment a (1965).
38. Id. at 631.
39. Id. at 632.
41. Id. at 7.
42. Id. at 6.
nally, there is often a professional organization that sets codes of ethics, and administers exams to those desiring to be members of the profession. According to Moore, these characteristics are set out on a scale up which professionals strive to advance in order to heighten their prestige. Some occupations do not meet all of these criteria, but because of their status with respect to other occupations, they are viewed as professions by the public.

A closer examination of the characteristics set out by Moore reveals that meeting all the criteria does not necessarily make one a professional. The fact that one's work is one's occupation, according to Moore, is just "a ticket of admission to the game." On the other hand, Moore claims that the commitment to a calling embodies some very important aspects of professionalism. He sees this commitment as involving "an acceptance of the appropriate norms and standards and an identification with professional peers and the profession as a collectivity." A professional insists on being set apart from the public, and is angered when a peer does something illegal. Further, identification of occupational interests often leads to a professional organization. This organization sets standards of behavior, thereby limiting entrance into the profession and embarrassment by peers. The fact that members of an occupation have enough common interests to feel a need for an association moves that occupation up on Moore's scale toward professionalism.

The most common characteristic of a professional is special education. The Federal Communications Commission defines a profession as an "occupation requiring either college graduation or experience of such kind as to provide comparable background." According to Moore, in many societies today formal educational qualifications are the rule for entrance into a profession. The curriculum involved in one's studies are often designed specifically for a particular profession. The length of study varies, although an arbitrary minimum might be set at a college bachelor's degree. The important aspect of a professional's education is not the degree itself, but that the training is so great that the common layperson cannot easily pick it up. It is this special skill that causes the services of the professional to be sought out. Thus on Moore's scale, the professional possesses not only the knowledge of an average

43. Id.
44. Id. at 5-6.
45. Id. at 7.
46. Id. at 8.
47. Id. at 9-10.
48. Id.
49. Id. at 11.
50. Id. at 16.
or even intelligent layperson, but also the special training needed to do things the average member of society cannot do.

According to Moore, many professions can also be seen as client-oriented. The client is the employer. The client places trust in the hired professional, and the professional then attends to the client's welfare and problems. The client initiates the relationship, and therefore the professional is dependent on the good opinion of the laity to find work. The ability of the professional to solve the client's problem is used to reassert authority. Although this "backward" relationship may create problems for the professional, the service provided maintains the importance of the professional in the eyes of the client.

C. IS A PROGRAMMER A PROFESSIONAL?

Whether professionalism is viewed as a ladder one must climb or as a cluster of attributes, the status of the programmer/analyst must be determined. First, it should be noted that the programmer under consideration earns a living as a programmer. While this may seem basic, it distinguishes the home computer owner who writes small programs for personal use.

The next step on Moore's scale is "commitment to a calling." This seems similar to Dean Pound's "pursuit of a learned art in the spirit of public service." Evidence of commitment to a calling in the data processing industry can be found in the new professional associations and their guidelines for "professional" conduct. In their journal The Communications of the A.C.M., the Association of Computing Machinery sets out a code of professional conduct and procedures for enforcement of the code. Moore states that a calling includes an acceptance of the appropriate norms and standards of a profession, and that the ACM code is an attempt to provide such standards.

The preamble of the rules states: "recognition of professional status by the public depends not only on skill and dedication but also ad-

52. W. Moore, supra note 40, at 87.
53. Id. at 97.
54. Id. at 88.
55. Id. at 92.
56. Id. at 7.
57. R. Pound, supra note 35, at 5.
59. Id.
herence to a recognized code of Professional Conduct." The rules call for integrity, competency, and professional responsibility by ACM members. In response to Dean Pound’s “spirit of public service,” Cannon 5 of the Code requires the ACM member to act for the advancement of human welfare. A major problem with this “calling” criteria is that neither membership in the organizations nor observance of their professional codes is mandatory. The computer industry is young, however, and these groups are working to make their standards prevail. Historically, professions have emerged as the demand for their services has grown in the lower classes. Clearly the computer industry is growing: computers are emerging as a part of everyday life. Thus the growth of data processing organizations seems to be in step with the growth of the industry itself.

This increased demand for computers throughout society has caused the industry to move up another step on Moore’s scale: the industry has become service and client-oriented. Typically, a programmer/analyst takes a problem presented by a client and tries to solve it. In designing a system to perform the desired function, the programmer uses intellectual judgment, discretion and authority. Clients are generally interested only in the end result reached by the programmer; they neither care nor understand how that result is reached. Thus, as in all professions, the programmer provides a necessary service to the client.

The most questionable aspect of computer programming in characterizing that occupation as a profession is the education of the programmer/analyst. Clearly programmers possess both “advanced education” and “special competence.” Whether a programmer receives a Computer Science degree from a university or attends a technical school, the training is both special and advanced. A higher standard of education, however, is problematic. The FCC calls for a college education or similar experience. Moore, however, suggests that a bachelor’s degree may be too low a minimum threshold for professionalism. While most programmers are college graduates, there are no universal training requirements. Educators have acknowledged a need for set national standards: those regulations that do exist are merely recommended.

In addition to this lack of curriculum uniformity, there is a shortage of college-trained programmers. It is projected that by 1990 there will be over 54,000 job openings for programmers, with only

60. Id. at 706.
61. Id.
62. W. Moore, supra note 40, at 56-57.
63. Id. at 11.
11,000 college-trained programmers available to fill them.\textsuperscript{66} This shortage is especially problematic in light of the fact that most of the employers offering these positions desire college graduates.\textsuperscript{67} To fill those 43,000 additional jobs, data processing managers will have to turn to the technical trade schools. This realization is troublesome. According to a Department of Education Report, at present there are 683 non-college programs for data processing in the United States, with an enrollment of 46,000.\textsuperscript{68} The tech schools do not attract the caliber of students that universities do. Most tech-school students are not interested in a four-year school.\textsuperscript{69} They are young and see computers as a way of avoiding an impoverished life.\textsuperscript{70} Employers must be concerned with the quality of education these students receive. A New York study reports that only twenty-five of the 230 programs offered by trade schools in that state provide modern instruction.\textsuperscript{71} While such studies have not been conducted in all states, one must wonder whether the results would differ elsewhere.

Quality of education may be a problem at universities as well. A study conducted by the Association for Computing Machinery reports that twenty-five percent of the undergraduate and thirty-three percent of the masters programs in this country do not meet ACM minimum standards.\textsuperscript{72} The Certificate of Data Processing exam, offered by the Institute for the Certification of Computer Professionals, has a success rate of only thirty-five percent.\textsuperscript{73} Furthermore, the test is not mandatory: only those who wish to take it do. One must question what would happen to the passage rate if the test were required of all data processors. Thus while all programmers receive education providing competence and skill beyond that of the average person, programmers with excellent university-level qualifications are rare.

Even after considering the various definitions of a "professional,"\textsuperscript{74} it is still unclear whether a programmer/analyst should be so considered. Clearly a programmer receives compensation and in that sense is not an amateur. The data processor works for a specific client and solves that client's problem. The programmer uses intellectual judgment and authority in designing a computer system. Professional as-

\textsuperscript{67} \textit{Id.} at 30.
\textsuperscript{68} Martin, \textit{supra} note 65, at 70.
\textsuperscript{69} Roman, \textit{From These Hallowed Halls}, COMPUTER DECISIONS 136, 142 (May 1983).
\textsuperscript{70} Martin, \textit{supra} note 65, at 66. Studies show that 80% of tech school students are under 30, 58% percent of them are women, and 28.5% are non-white.
\textsuperscript{71} \textit{Id.} at 70.
\textsuperscript{72} \textit{Id.} at 74.
\textsuperscript{73} \textit{Id.} at 76.
\textsuperscript{74} \textit{See supra} text accompanying notes 34-55.
associations have developed and have established codes of ethics. With respect to these criteria, the programmer seems to be a professional. The educational criterion, however, is problematic. The programmer has the "special competence" required by the Restatement definition, and the "advanced training" required by the Webster's Dictionary definition. As desired by Dean Pound, the programmer pursues a "learned art in the spirit of public service." But the Federal Communications Commission definition emphasizes a college education, and Moore is reluctant to accept even a bachelor's degree as sufficient education when defining "professional." These more stringent definitions of "professional" seem to be the norm in our society. Medicine and law, two traditional professions, both require graduate degrees. Accounting and engineering, occupations that have just recently been classified as professions, call for university degrees.75 Furthermore, each of these four occupations is licensed through standardized testing.76 Thus it is arguable that the programmer/analyst is not a professional: the number of college-educated programmers is simply too small. This situation is changing however. Organizations are calling for certification of data processors and have proposed model curriculum.77 Additionally, existing university programs are growing and new ones are arising annually.78 Thus if higher education is viewed as a mandatory requirement of a true professional, then the programmer/analyst is not there yet. The computer programmer, however, meets most of the criteria of a professional, and must be considered well on the way up Moore's scale.

D. DEVELOPING THE HIGHER STANDARD OF CARE

To conclude that a programmer/analyst should be held to a higher standard of care would require a determination of what that standard should be. The standard for professional negligence requires "one who practices a profession . . . to exercise the skill and knowledge normally possessed by members of that profession in good standing."79 The courts have stated clearly that a professional is not infallible.80 According to Prosser, one who has knowledge, skill, or even intelligence supe-

75. According to the OCCUPATIONAL OUTLOOK HANDBOOK, supra note 64, "[a] bachelor's degree in engineering is generally acceptable for beginning engineering jobs." Id. at 49. The HANDBOOK also states that most public accounting firms require at least a bachelor's degree in accounting, while others require a master's degree. Id. at 24.

76. Over 250,000 accountants are licensed in some way. Id. at 24. While most engineers take the engineer-in-training exam, other testing depends on the type of engineering one pursues. Id. at 49.


78. Simpson, supra note 66, at 70.


rior to an ordinary man must act in a way that is consistent with that knowledge. Prosser states that a professional must act like the average member in good standing of his profession. This standard will be modified, however, if the professional holds himself out as having skills either better or worse than average. When examining the duties of a professional it is normal skill, not normal care, that is required. The act itself is examined to determine whether the professional applied the average skill of those possessing the same education and experience. It must be realized that one's training apprises one of the special risks involved in a chosen profession, and that therefore one must act reasonably considering those challenges. Thus the professional negligence standard is merely the reasonable person standard after taking into consideration the professional's training. At first glance it appears that the professional standard of care does not create many extra duties.

An examination of medical malpractice will illustrate the basic professional negligence standard and the ways in which it can be strengthened. When a physician takes a case, there is an implied representation that the physician "possesses and will exercise the reasonable or average degree of learning and skill ordinarily possessed by physicians or surgeons of ordinary and average learning." A physician who is a specialist must adhere to a standard commensurate with his or her extra ability and training.

The rule used to determine the level of skill possessed by the "average physician" has developed over time. The old rule looked at the custom in the locality of the physician in question. This rule was established to protect physicians in small towns. It was thought that those practicing in rural areas would not be exposed to new techniques in the profession as quickly as physicians in major cities. Furthermore, it was felt that because the practice of medicine requires the use of a great deal of judgment, a physician must be free to operate as long as accepted procedures are followed. This rule, while still intact to some extent, is changing. Some cases have held physicians liable for follow-

82. Id.
83. Id. at 166.
84. Curran, supra note 6, at 537.
85. Id. at 538.
90. Id.
ing local custom where the custom itself was negligent.\textsuperscript{91} In fact, there is evidence that the locality rule is eroding completely. The emphasis has turned toward specialization, with each specialty having a national standard of care.\textsuperscript{92} Another trend is movement toward shifting the loss from the injury to the health-care provider who then distributes it to the public.\textsuperscript{93}

Another traditional burden placed on the physician comes from the doctrine of informed consent. Before a physician performs any treatment on a patient, the physician must have the patient’s “informed consent.” This means that the physician must communicate to the patient any information a reasonable person would want to know before undergoing the treatment in question.\textsuperscript{94} “[E]very human being of adult years and sound mind has a right to determine what shall be done with his own body.”\textsuperscript{95} Because the patient must bear the expense, pain, and suffering from the treatment, the patient’s right to know should be set by law. Because physicians have been found to have a bias toward underdisclosure, a legal standard would better protect the patient.\textsuperscript{96}

There are four characteristics of the doctor-patient relationship that create the need for informed consent. First, the patient is uneducated in the medical sciences. Second, as an adult the patient has the right to decide whether to submit to treatment or not. Third, the only way an adult can consent effectively is if the consent is informed. Fourth and most importantly, the patient completely relies on the physician to provide the information used to make the decision.\textsuperscript{97} This doctrine places an affirmative duty on the physician and, along with the locality rule and the professional negligence standard, creates specific standards of care for the physician.

The courts have also made it easier for injured patients to bring malpractice actions. Because of the statute of limitations, the time that a claim accrued is often the central question in a case. Ordinarily, a cause of action accrues when the wrong occurs.\textsuperscript{98} This is problematic in

\textsuperscript{91} Id. at 18 (quoting Favalora v. Aetna Casualty and Sur. Co., 144 So. 2d 544, 550-51 (La. Ct. App. 1962)).
\textsuperscript{92} Michaud & Hutton, supra note 88, at 730.
\textsuperscript{93} Hirsh, Judicially Imposed Standard of Care—Prophecy in Medicine, 1981 MED. TRIAL TECH. 1, 2.
\textsuperscript{94} Note, California Supreme Court Expands Informed Consent Doctrine, Physicians Have a Duty to Obtain an Informed Consent: Truman v. Thomas, 1980 B.Y.U. L. REV. 933, 936.
\textsuperscript{96} Annot., 88 A.L.R.3d 1008, 1017 (1978).
\textsuperscript{97} Note, supra note 94, at 939.
\textsuperscript{98} 2 Witken, California Procedure 2d 1116 (1970).
the malpractice area, however, as the treatment of a specific ailment may continue for months or years. The courts have pointed out that it would be ridiculous to require a person to interrupt treatment in order to serve a summons on the physician. Moreover, because the patient must depend on the physician for information, acts of malpractice may not be readily apparent to the potential plaintiff. Thus because of the nature of the physician-patient relationship, potential plaintiffs are often unable to make timely claims. To deal with this problem, the courts have developed the continuous treatment doctrine. The doctrine provides that, "when the course of treatment which includes the wrongful acts or omissions has run continuously and is related to the same original condition," accrual will be deemed to occur after the treatment is completed. Thus fewer claims are barred for being untimely. It is felt that where the patient must depend entirely on the physician's judgment, the continuous treatment doctrine is more fair.

Application of the continuous treatment doctrine has been extended to other professions. In *County of Broome v. Vincent J. Smith, Inc.*, the doctrine was applied to an architect. The court stated that the doctrine could be applied to professions involving confidential and complex relationships. The effect of the doctrine has been to allow malpractice actions to accrue later than they otherwise would, and therefore to make enforcement of professional standards of care easier.

E. WHETHER PROGRAMMERS SHOULD BE HELD TO A HEIGHTENED STANDARD OF CARE

Three cases have considered whether professional negligence can be applied to the programmer/analyst. *F & M Schaefer Corp. v. Electronic Data Systems, Inc.* held that a malpractice claim may be brought against a computer supplier. The court found that the defendant may have negligently misrepresented its system and that the contin-

---


103. Id.

104. Id. at 1002.

105. The theory has also been applied to lawyers. See Grago v. Robertson, 49 A.D.2d 645, 646, 370 N.Y.S.2d 255, 259 (N.Y. Sup. Ct. 1975) (a cause of action against an attorney accrues when representation concerning a certain transaction ends).

uous treatment exception to the statute of limitations was applicable. The court left it to the jury to decide whether a professional-client relationship existed. The applicability of the continuous treatment doctrine was based on the determination that the computer supplier "may be compared to architects for this purpose." The Schaefer holding is unclear, however, as the judge did not rule on the existence of the required professional-client relationship. Thus the decision does not show acceptance of professional standards of care for computer programmers.

In Chatlos Systems, Inc. v. National Cash Register Corp., the court considered in a footnote "the novel concept of a new tort called 'computer malpractice' premised upon a theory of elevated responsibility on the part of those who render computer sales and service." The court stated:

Plaintiff equates the sale and servicing of computer systems with established theories of professional malpractice. Simply because an activity is technically complex and important to the business community does not mean that greater potential liability must attach. In the absence of sound precedential authority, the Court declines the invitation to create a new tort.

The court saw this case as a contract claim for the sale of goods, not as an action for negligent services. The plaintiff purchased a computer system and the problems caused by its failure to work were, according to the court, covered by a warranty, not by the services of a professional.

A third case, Triangle Underwriters, Inc. v. Honeywell, Inc., is similar to Chatlos. The court ruled that the defendant was a seller and not a professional. "It was never contemplated that Honeywell would undertake the continuous running of the data processing system for Triangle." The court stated that Honeywell merely manufactured and sold equipment to Triangle. With respect to the continuing treatment question, the court ruled that "New York law provides generally that a manufacturer's efforts at repair subsequent to delivery do not extend the contract statute of limitations." According to the court, previous application of the exception "has depended upon that particular relationship of trust and reliance that exists between a lay plaintiff and

107. Transcript of hearing Nov. 15, 1977 at 175-76.
108. Id.
110. Id. at 740 n.1.
111. Id.
112. Id. at 742.
113. 604 F.2d 737 (2d Cir. 1979).
114. Id. at 745.
115. Id.
116. Id.
the professional defendant." The court seemed to feel that the client had not placed the necessary trust and reliance in the defendant. It is important to note, however, that the court ruled that Honeywell was merely a manufacturer and seller. In such a situation, there is no professional relationship and the continuous treatment doctrine does not apply.

Presently, no court has recognized a new tort of "computer malpractice." In each of the cases discussed, the required professional relationship was found not to exist. In two of the cases the court considered the computer defendant merely a manufacturer and seller. In Schaefer the court did compare EDS to architects, but still refused to consider it a profession. It is arguable, however, that these courts did not view the situation correctly. The cases all included both the sale of hardware and the design and sale of software. With respect to malfunctioning hardware, classification of the defendant/supplier as a mere manufacturer seems valid: the hardware is mass-produced by machinery as are most products. The design of the software, however, is different. In the case of software, the client brings the computer company a specific problem to be solved. The data processors take the information and, using their special education and skill, design a system that will perform the desired task. The process is unique, not repetitive. The programmer uses intellectual judgment and authority throughout. The client, unable to solve the problem alone, entrusts the programmer with it. Thus if the courts had viewed the computers supplied as involving two distinct products—hardware and software—then a professional relationship with regard to the software design would have been found. Given the existence of the required relationship, the courts should have found that the continuous treatment doctrine applied and awarded recovery under a theory of computer malpractice.

IV. WHAT SHOULD BE DONE WITH THE PROGRAMMER/ANALYST

A. CONTRACT LAW

After determining the legal theories that might be used to impose extra duties on the computer programmer, each must be examined to determine how it would be adapted to fit the computer context. The two contractual theories might be used in similar ways to force a computer programmer to make the necessary disclosures to the client. Assuming that the programmer is a fiduciary, it could be argued that one of the extra duties required should be the duty to inform the client about the need for documentation. This would allow the businessper-

117. Id.
son to make an informed decision regarding the purchase of documentation. It is unclear whether a court would classify a programmer as a fiduciary. As noted previously, a fiduciary relationship is one in which one party places so much trust in the other that domination results.

There are two typical contexts in which a fiduciary relationship exists. The first is the corporate setting, in which all the officers of a corporation have a fiduciary relationship to the shareholders. The other is the trustee-beneficiary relationship. There the trustee works directly with the beneficiary's money and must manage it in the beneficiary's interest. While it can be argued that a client places complete trust in the programmer, these classic fiduciary relationships involve a more direct interest. It seems that the programmer-client relationship should constitute a fiduciary relationship. If the types of relationships that are presently considered fiduciary are any guide, however, convincing a judge to impose a fiduciary duty on a programmer may be difficult.

Approaching the problem under a quasi-contract theory, it must be argued that the duty of fair dealing requires a programmer to make disclosures about documentation to the client. This argument can be made by comparing the programmer-client situation to the previously discussed insurance cases. There, good faith required acceptance of all reasonable settlements. The cases held that there is an implied covenant in every contract that neither party will do anything to injure the right of the other to receive the benefits of the contract. In the computer context, lack of documentation injures the benefit derived by the client from the program. The client must go to the original programmer for help. Just as a duty to accept reasonable settlements was imposed on insurance companies, good faith would seem to require the programmer to disclose the need for documentation to the client. The requirement of good faith and fair dealing seems especially appropriate to deal with the programmer-client relationship. Unlike fiduciary duty, which applies only to limited situations, the good-faith requirement applies to all contracts.

B. TORT LAW

Although it has not yet been held that the programmer/analyst is subject to a professional standard of care, it is time for such a holding. With this realization, it must be determined which aspects of malpractice should be applied to the programmer/analyst. The basic standard of care for professionals would require the programmer to exercise the skill and knowledge of the average member of the profession. Expert testimony would be required to determine the level of skill required. The locality rule seems inappropriate. Given modern modes of travel and communication, each programmer/analyst should be expected to
keep up with changes in the field. Furthermore, if a model curriculum for programmers were adopted, all programmers would have similar educations prior to entering the field. Thus a national standard should be adopted, requiring programmers throughout the country to possess the same minimum skills.

Beyond the basic professional standard of care, imposing additional requirements on the programmer/analyst may be appropriate. Physicians are subject to the informed consent doctrine. Before administering treatment, the physician must inform the patient of the risks involved in the proposed treatment and the possible alternatives. There are four reasons for placing this requirement on the physician: the patient is unlearned in the medical sciences; an adult has the right to decide whether to submit to the treatment; effective consent to treatment must be informed; and the patient relies on the physician for information. This notion of needed information is key in the programmer-client context. Generally the client has no idea what the program ordered consists of. If, upon being hired, the programmer/analyst were required to explain to the client what a program consists of, what purpose each part serves, and what the risks of not having each are, then the client could make an informed decision. While many computer clients do not want their programmers to write documentation, at least the decision would be the client's. The reasons for putting the informed consent burden on physicians all seem to apply to the computer context: the average client knows little about computer programming; the client should have the right to decide what features will be purchased and what risks will be taken; the client's decision regarding the purchase of the program must be informed; and the client relies solely on information from the programmer in making a choice about a computer system. In light of these considerations, a type of informed consent doctrine should be imposed on the programmer/analyst.

After concluding that a strict standard of care for programmers should be required, it follows that the continuous treatment theory should apply in order to make computer malpractice claims easier to bring. The client completely relies on the professional's judgment. At least one case has suggested that the theory can be applied to relationships that are confidential and complex. Since a programmer/analyst undoubtedly acquires confidential information about the client when designing a system, these criteria seem to be present. The alternative justification for the continuous treatment doctrine is awkward to apply in this context. Generally the statute of limitations tolls when a wrong occurs. In the computer context, there is constant output that is af-

fected by any mistake. Thus most cases of negligence would be easily recognizable by the client. It is only with a lack of documentation that complications arise. Under the continuous treatment theory, if the treatment that includes the wrongful act is continuous and for the same original condition, then the cause of action accrues after the treatment has ended. In the programmer-client context, however, treatment of the original program could continue for decades. In that case, when the programmer leaves, the statute of limitations would have expired. Thus perhaps the statute should begin to run when the wrong is discovered. In most cases involving programming negligence this would be shortly after the wrong occurred. In those rare cases where the negligence is not easily recognizable, the client is protected. Thus the continuous treatment theory is not fully applicable to the computer programmer standard of care.

There remain certain actions that must be taken to assure adequate service from programmers. First, it is imperative that a standardized course of instruction with minimum standards be adopted. Organizations such as the Data Processing Management Association have proposed model curriculum. Something must be agreed upon by the industry and put into effect. This will guarantee that each programmer will have the required skills before receiving a computer science or programming degree. To further standardize programmers' qualifications an association similar to the legal bar should be established to administer an exam to persons wishing to enter the computer field. The Committee for the Certification of Computer Professionals is already attempting to establish such an exam. Finally, a standardized code of ethics for programmer/analysts should be adopted. Again, existing professional associations have developed such codes; one should be adopted as mandatory.

While licensing is the best way to regulate programmers, licensing has its faults. A major problem is the cost involved. The bureaucracy necessary to administer and regulate such tests and requirements would be huge and extremely expensive. Furthermore, standardized testing causes rigidity in the profession's training. As with many law schools, students will be taught what is needed to pass the test. Experimentation and scholarly thought would be inhibited. Finally, as with the legal bar and the Medical Board of Quality Assurance, an organization testing programmers would be run by the profession itself. Having programmers test and set standards for other programmers brings into question the value of the whole process. Despite these criticisms, there is no better system for preventing harm. All other legal methods merely react to wrongdoing after the fact. While this is of course necessary, early prevention is a noble goal.
CONCLUSION

With the prevalence of computers in our society, the law must address the new problems they create. Because the average person knows little about computers, those in the industry can easily take advantage of uninformed consumers. Since custom designed programs are now commonly used by small businesses, the relationship between the programmer/analyst and the client must be dealt with immediately. Two theories within contract law represent potential ways of creating extra duties of disclosure for the programmer—fiduciary and quasi-contract theories. Such duties of disclosure would better ensure that the client had the information needed to bargain effectively with the programmer. Alternatively, a professional standard of care requiring disclosure might be created for the programmer/analyst. The programmer/analyst seems to fit well within almost any definition of a "professional"; at the very least the programmer/analyst is rapidly approaching that stature. Therefore the courts should adopt a stricter standard of care for the programmer, including a duty to inform the client of the risks and advantages of the system provided. Furthermore, uniform requirements for education, ethics and licensing must be adopted. Whether such a comprehensive standard is adopted, or a duty of disclosure is implied through contractual theories, the client will no longer be at the mercy of the programmer.

Sam Kraemer