OCCASION

This publication has been made available to the public on the occasion of the 50th anniversary of the United Nations Industrial Development Organisation.

DISCLAIMER

FAIR USE POLICY

CONTACT

Please contact publications@unido.org for further information concerning UNIDO publications.

For more information about UNIDO, please visit us at www.unido.org

UNITED NATIONS INDUSTRIAL DEVELOPMENT ORGANIZATION
Vienna International Centre, P.O. Box 300, 1400 Vienna, Austria
Tel: (+43-1) 26026-0 · www.unido.org · unido@unido.org

20459

Distr.
LIMITED

IPCT.190(SPEC.)
23 November 1993

UNITED NATIONS
INDUSTRIAL DEVELOPMENT ORGANIZATION

ORIGINAL: ENGLISH

LEGAL PROTECTION AND INNOVATION IN THE SOFTWARE INDUSTRY*

Prepared by

Carlos M. Correa
UNIDO consultant

* The views expressed in this document are those of the author and do not necessarily reflect the views of the Secretariat of UNIDO. This document has not been edited.

V.93 90458

TABLE OF CONTENTS

ANNEXES

INTRODUCTION

The software industry constitutes today the most dynamic segment and the driving force of the informatics sector. Said industry has grown and internationalized at a rapid pace during the 1980's, with global revenues exceeding U$S 65 billion at the end of the decade (OTA, 1992, p. 94).

The development of the software industry has given rise to one of the most complex issues in contemporary law: how and to what extent should the law protect computer programs. They are functional works, produced and used under specific technical rules, which often operate in connection with other programs that may belong to other parties. Easy to copy but hard to develop, computer programs pose a serious challenge to the adaptability of existing intellectual property law. Legal issues relating to software are complicated and many still remain unresolved.

A "first generation" of case law and legislation was concerned with the protection of software as such.[1] Courts and parliaments in various countries faced the basic problems of copyrightability of computer programs in source and object code and dealt with several objections to the extension of copyright principles to this new field (Correa, 1990a). Industrialized countries, following the lead of the United States, adopted the copyright approach, albeit with some significant differences.[2] Some developing countries (e.g.India) adopted the same form of protection, but in many cases it was the result of actions undertaken by the United States under section 301 of its Trade and Tariffs Act. It is clear today that copyright constitutes the main - though not the only[3] - framework for software protection and recent international negotiations are likely to provide a firm basis for this solution. This does not imply, however, as pointed out below (section 1. a) that the debate on the form of protection is definitely closed.

"Second generation" cases gave copyright protection as granted and entered into other difficult issues, particularly how the traditional expression/idea dichotomy applies to software with respect to its internal organization and structure and to the so-

---

[1] "Computer program" and "software" shall be used in this paper as synonymous for practical purposes only.

[2] Illustrative of diverging solutions were the amendments introduced to the Japanese and French laws in order to accommodate the protection of computer programs.

[3] Software is not only protectible in many countries under copyright but also under trade secrets law. There is also a trend in the United States to increasingly admit the patentability of algorithms and computer programs, as illustrated by the recent "Arrhytmia Research" case. For a discussion of the referred trend, see Samuelson, 1992.

called "look and feel".[4] Case law of this type has almost exclusively arisen in industrialized countries, where most software production is concentrated.[5] These problems have also been the subject matter of at least one international regulation, in the case of the EC Software Protection Directive approved in 1991 after intense debate. Notwithstanding the importance attached by the industry and legal experts to finding appropriate responses to said issues, a great degree of uncertainty still exists, especially after recent case law in the United States.[6]

Debate and contention on legal protection of software have not been the result exclusively of inadequacies in the legal system. The solutions eventually given affect innovation and competition in the industry, may erect entry barriers for new potential producers and influence the rate of diffusion of computer technology . Thus, many developing countries have feared that a "strong" protection might lead to an imbalance in producer-user relationship and to prices too high for local users.[7] Divergences have also been marked among software/hardware producers in industrialized countries, particularly as regards to the admissibility of "reverse engineering" as a method to develop competitive products.[8] Concerns have been voiced in connection with the impact of over-protection on the rate of innovation in major producer countries as a matter of public interest.[9]

This paper addresses recent international trends in software protection and some of the unresolved issues in "second generation"

---

[4] The "look and feel" issues refer to the protection of elements in the menu hierarchies, command structures, key sequences and other aspects of a program's user interface.

[5] Despite the efforts of some developing countries, like India, their share of the global software market is still insignificant (around 2%). See on this issue Correa, 1990b, and Schware, 1992.

[6] This country, which maintains an uncontested leadership in software production and trade (with around 60% of the world market), has had an important role as standard-setting in the area of software protection. The referred judicial cases are considered later in the text.

[7] Several developing countries (e.g. Brazil, Thailand) resisted U.S demands to improve legal protection for software for reasons related to local industrial policies or to technology diffusion. For an analysis of economic aspects related to software protection, see Mody, 1989.

[8] The main battle on this issue was prompted by the EC initiative to adopt a Software Protection Directive. A number of firms willing to counterbalance the influence of major industry players constituted the "European Committee for Interoperable Systems" (ECIS), which actively participated in the debate (see section 3, below).

[9] See, for example, Business Week, 1989.

cases. Its main aim is twofold: on the one side, to provide an overview on emerging international standards that may influence legislation and practice worldwide; on the other, to deal with some critical legal issues for software producers, mainly in developing countries. Computer technology requires a special expertise and employs technical terms that lawyers find hard to understand. Conversely, software producers are not familiar with legal terms and with the implications of the evolving law that apply to the products they develop. This document attempts to fill that gap. The paper is not intended to deal, except to the extent necessary for analyzing recent international trends, with "piracy" issues, i.e. those relating to unauthorized slavish copying of third parties programs. Notwithstanding the importance of this problem, it is better known than the issues of "second generation" cases and, as mentioned, it is relatively well covered by existing legislation and case-law.

Section 1 reviews the outcome (as relevant to software and data bases) of the negotiations held in the framework of the Uruguay Round in order to adopt an Agreement on Trade Related Aspects of Intellectual Property Rights (TRIPs). Said negotiations, which started in 1986, have produced a draft Agreement (submitted by GATT Director General on December 21, 1991) which may eventually become enforceable in case the Round's negotiations are unblocked and successfully completed. The draft contains the most far-reaching intellectual property international agreement ever reached in terms of its comprehensiveness and detail of proposed minimum standards.[10] Consideration is also given to work undertaken within the World Intellectual Property Organization in order to prepare an eventual protocol to the Berne Convention.

Section 2 briefly deals with the expression/idea dichotomy problem in the context of software protection according to US, European and Japanese law. This section is intended to provide elements for the comprehension of section 3, which deals with "reverse engineering" as a process to analyze and evaluate an existing product or technology in order to find its underlying concepts, structure and functionalities. The latter section also considers various policy issues and procedures of software development which may avoid problems stemming from the eventual application of copyright.

Section 4 contains the main conclusions of the study.

1. INTERNATIONAL DEVELOPMENTS

As mentioned, international developments concerning software protection have taken place within two fora: GATT (Uruguay Round)

---

[10] The draft Agreement includes minimum standards, which in most cases supplement international conventions, on all areas of intellectual property (copyright, patents, industrial designs, trademarks, indications of origin, designs of integrated circuits, "undisclosed information") except breeders' rights and utility models.

and WIPO. While GATT negotiations have led to a draft text that might be adopted as soon as the Uruguay Round concludes, the latter are still at a more preliminary stage. Another important development (which is not examined in detail here) has been the approval of the referred EC Directive on the Legal Protection of Computer Programs, which is likely to substantially influence legislation in many countries, particularly in view of the elaborated solution reached on issues dealt with in section 3 of this study.

1.1 TRIPs Draft Agreement

Section 1 of the TRIPs draft Agreement deals with "copyright and related rights". It contains six articles deemed to supplement existing international conventions on the matter, particularly the Berne and the Rome Conventions. However, while in some cases the proposed texts imply additional protection, in others they set out standards that are below those established by said conventions.

Unlike the section relating to patents, negotiations on the copyright area have been characterized by a North-North confrontation on a number of issues that concern the fundamentals of intellectual works' protection. The main differences have arisen from the diverging views prevailing in common law countries on the one side and in continental law countries on the other, on the concept of author as applied to various works and on the scope of protection accorded to them. The title of Section 1 (Part II) itself reflects a compromise between the U.S. and European conceptions on the nature of rights relating to phonogram producers.[11] The relevant parts of the TRIPs draft Agreement are reproduced in Annex 1.

a) Software as literary work

The recognition of copyright as a main modality for the protection of computer programs has been a major objective of industrialized countries in TRIPs negotiations. Article 10.1 provides that "computer programs, whether in source or object code, shall be protected as literary works under the Berne Convention (1971). Several comments on this provision are relevant here.

First, the most suitable form of protection for software has been the subject matter of an intense, and still open, debate. Many experts and courts have sustained the applicability of the Berne Convention to software, even in the absence of an express reference to that effect in its text (WIPO, 1992b , p. 49-50). However, a good part of legal and economic literature has questioned the appropriateness of copyright protection for functional work such as software (see Correa, 1990; Mody, 1989), and some influential institutions have advocated the development of a "new category of

---

[11] Under U.S. law such rights are deemed to be a chapter of "copyright law", while in Europe they are considered as a part of a separate category of "neighbouring rights".

law" (Office of Technology Assessment-OTA, 1986, p. 14). The treatment of software as a part of a TRIPs Agreement has also been criticized as inconvenient in view of the present status of the debate on the matter (Uchtenhagen, 1990).

Second, while copyright as a basis for protection of computer programs has been accepted by a large number of countries (in part as a result of U.S. government action under Section 301 of the Trade and Tariffs Act) many have been reluctant to consider software as a literary work.[12] The main implication of this assimilation is the extension of the term of protection, since other aspects (e.g. number and type of permitted copies for a legitimate user) need to be addressed, taking the specificity of computer programs into account.[13]

Third, Article 10.1 applies to computer programs "in source or object code", but is silent with respect to cases where they are embedded in microelectronic devices. Likewise, no specific rule is established in connection with reverse engineering, one of the most controversial issues in this field.

Fourth, protection via copyright does not necessarily exclude other forms of protection. Computer programs are deemed in some countries to be protectible as trade secrets. In the United States, moreover, as mentioned, a trend towards the acceptance of patents on algorithms and computer programs is progressing today.

b) Rental rights

Article 11 contains one of the main innovations in the copyright section of the draft Agreement. It introduces as a minimum standard the recognition of "rental rights" in respect of "at least computer programs and cinematographic works". They confer the title-holders the "right to authorise or to prohibit the commercial rental to the public of original or copies of their copyright works".[14]

The origin of Article 11 can be traced in recent U.S. legislation (which the adopted text mirrors to a large extent) that specifically recognized rental rights for computer programs and

---

[12] In France, for instance, computer programs were granted a 25-year protection as in the case of works of applied arts (see Correa, 1990).

[13] Many countries, including the United States, have explicitly dealt with such issues in legal reforms introduced during the last decade.

[14] Despite some proposals made during negotiations, the commented text does not provide for an alternative right to obtain an equitable remuneration for rental.

sound recordings (but not for visual works).[15] While these rights entail an exception to the "first sale" doctrine that prevails in common-law countries, the exclusive right to rent may be deemed as a part of full distribution rights as recognized under the law of some countries (see EC Commission, 1988 p. 146). The EC Commission has also felt the need to propose an EC Directive to harmonize rental rights belonging to authors, performing artists in respect to fixations, phonogram producers and producers of the first fixation of cinematographic works.[16]

To sum up: the TRIPs draft Agreement introduces exclusive rental rights for the first time in an international agreement though limited to computer programs and phonograms (and, subject to a broad exception, to cinematographic works). The adopted solution mirrors to a large extent the situation in U.S. copyright law, but on a permanent basis and lacking specific exceptions for non-commercial acts. The exceptions and the limitations of the proposed provisions reflect the still pending differences among developed countries on copyright concepts. However, since the new standards are "Berne-plus" and "Rome-plus", they are likely to lead to legislative changes in many developing as well as developed countries.

c) Term of protection

Article 12, relating to the term of protection, seems to just reiterate the Berne standards. However, it settles a difficulty encountered in the application of the said Convention in countries whose legal systems allow a juridical person (e.g. a corporation) to be considered as an "author"[17]: the referred provisions allude to the term "calculated on a basis other than the life of a natural person".

In addition, the commented Article makes it clear that failing publication of the protected work within fifty years from the making of the work, protection shall extend for fifty years from

---

[15] See the Record Rental Amendments Act and the Software Rental Amendments Act. While the TRIPs Agreement will establish, if finally approved, a permanent minimum right, the referred legislation is openly temporary (it shall expire on 1 October 1997) due to the U.S. Congress' unwillingness to create a perpetual exception to the "first sale" doctrine (Peterson and Makay, 1992). Extension of rental rights to movies has been refused in the United States.

[16] For an analysis of this draft Directive, see Von Lewinski, 1991. It should be noted that in contrast to Article 11, said proposed Directive would allow member countries to substitute exclusive rental rights by the right to obtain an equitable remuneration. It should also be recalled that rental rights are recognized for computer programs by the recently adopted EC Directive on the matter.

[17] Those cases have been dealt with under Article 7.3 of the Berne Convention, which refers to "anonymous or pseudonymous" works.

the end of the calendar day of the making.

Previous drafts of Article 12 included an explicit reference to computer programs. It was deleted once the assimilation of said programs to literary works was incorporated in Article 10.1. That assimilation makes a specific rule on duration of rights for software irrelevant.[18]

1.2 Developments within WIPO

WIPO convened in November 1991 a Committee of Experts in order to consider the preparation of a protocol to the Berne Convention for the Protection of Literary and Artistic Works. The objective of the proposed protocol is to clarify Berne rules or to establish new rules in cases where there may be doubts on the current scope of application of the Convention. The need for the protocol is grounded on the existence of divided opinions and national legislative approaches on a number of issues.

One of the issues pointed out as generating differences is software protection. The preparatory memorandum submitted by the WIPO Secretariat (WIPO, 1992a) presents and discusses the doubts expressed in connection with the protection of software via copyright. WIPO's document affirms, however, the applicability of copyright and proposes to explicitly include in a possible Protocol the obligation to protect computer programs like other literary and artistic works and to clarify that "computer programs" is deemed to include operating systems and application programs, whether in source or object code.

The referred memorandum (which is included as Annex 2 in its relevant parts) stresses that copyright law does not extend to algorithms, ideas, procedures, methods or concepts. It further states that decompilation (see section 3 below) should be permitted but only when necessary to obtain information to independently create programs that would interoperate with the original program, and provided that the new program is not substantially similar in its expression to the original one. This suggestion seems inspired by one of the early drafts of the EC Directive,[19] which restricted decompilation for purposes of developing programs compatible with those being decompiled. The final version of the Directive however, as approved on May 14 1991, eliminated such a restriction legitimizing decompilation to achieve interoperability "with other programs".

The convened Committee of Experts did not reach consensus either on the need for including a specific rule on computer programs in a possible protocol, or on the terms to be eventually

---

[18] Some countries (France, as mentioned, and Brazil) while adopting copyright for computer programs, limited its duration to twenty-five years.

[19] Draft by the EC Commission of October 18, 1990.

used, particularly as regards to the admissibility of decompilation. There was, however, a clear understanding that protection by copyright only concerns the expression and not the ideas, methods, procedures or underlying concepts, while some participants stressed that the same dividing line applies with respect to programming languages and rules (WIPO, 1992b).

No further substantive work has taken place on a possible protocol to the Berne Convention after the referred meeting of the Committee of Experts. The Assembly of the Berne Union decided in September 1992 to go ahead with the preparation of said protocol on the basis of a ten point list, one of which relates to "computer programs".[20]

1.3 Implications of International Developments

The TRIPs Agreement, if finally adopted, shall establish as a minimum universal standard that computer programs are

- Protected under copyright in accordance with the Berne Convention (1971);

- Assimilated to "literary works" whether in source or object code.

The application of rules as laid down by both the Berne standards and TRIPS to computer programs would have a number of implications, some of which are straightforward while others are subject to interpretation. One of the clear implications relates to the term of protection, which could not be in any case less than fifty years. Another one relates to the granting of "rental rights". Aspects which would be subject to interpretation and for which no clear international standard is yet set forth include:

- The applicability of "moral rights" to computer programs' authors;[21]

- The legitimacy of reproduction of a computer program for private use. How many copies could legally be made and for what purposes?;[22]

- The scope of the right to adapt a program for use on a

---

[20] See document WIPO B/A/XII/2, September 29, 1992.

[21] It should be noted that the TRIPs Draft Agreement explicitly excludes from its scope "moral rights" as provided by article 6bis of the Berne Convention. The relevance of such rights in the case of computer programs has been extensively discussed and in some cases (e.g. France) the legislation has abrogated them given the special nature of the protected works.

[22] This point has been clarified in many national laws, such as in the United States, Japan, Brazil, etc.

specific hardware or with a different computing language;

- The admissibility of decompilation and reverse engineering.

The TRIPs Draft Agreement falls short in providing standards on these issues. This means that they need to be addressed at the national level. Those issues might also eventually be the subject matter for negotiations on the establishment of a protocol to the Berne Convention as proposed by WIPO, if progress on the drafting of said instrument is finally made.

2. PROTECTION OF EXPRESSION AND IDEAS IN COMPUTER PROGRAMS

It is generally agreed that copyright protects "only the form of expression of ideas, not the ideas themselves. The creativity protected by copyright law is creativity in the choice and arrangement of words, musical notes, colors, shapes and so on" (WIPO, 1988,p. 209). This principle, however, has been implemented in different ways in various countries, which eventually leads to different results in close cases. Protection may, thus, extend not only to literally similar elements but also to non-literal elements if they are "comprehensive" relative to the overall texture of the work (Geller, 1991, p. INT. 33-34).

The Berne Convention is silent on this dichotomy, and therefore leaves national legislation certain room, within the limits of copyright principles, to draw the dividing line. Article 9.2 of the TRIPs Draft Agreement has instead opted for an explicit rule according to which "copyright protection shall extend to expressions and not to ideas, procedures, methods of operation or mathematical concepts as such".

Article 9.2 was introduced as a counter proposal to the Japanese demand to include an exception of this kind in Article 10. The Japanese text, however, was only related to the protection of computer programs and explicitly referred to the non-protectibility of algorithms. While this latter solution would have been restricted to such programs, the finally proposed text is applicable to all areas of copyrights and related rights.

In the following paragraphs the treatment of the expression/idea dichotomy in United States, Japan and Europe (based on the EC Directive on the Legal Protection of Computer Programs) is briefly reviewed. The analysis intends to highlight which is the dividing line separating protected and not protected elements in computer programs under the indicated legal systems.

2.1 United States

The basis for the expression/idea distinction in US law is Article 102 (b) of the Copyright statute, according to which protection does not extend to any

> "... procedure, process, system, method of operation, concept, principle, or discovery, regardless of the form in which it

is described, explained, illustrated or embodied".

While considering this issue in connection with computer programs on occasion of the 1976 amendment to the Copyright Act, the House Report stated:

> "Some concern has been expressed lest copyright in computer programs should extend protection to the methodology or processes adopted by the programmer, rather than merely to the 'writing' expressing ideas. Section 102(b) is intended, among other things, to make clear that the expression adopted by the programmer is the copyrightable element in a computer program, and the actual processes or methods embodied in the program are not within the scope of the copyright law".[23]

The application of this principle to a functional work as a computer program has not been an easy task. Case law has not reached a clear solution yet; moreover, recent cases illustrate contradictory views on the extent of protection.

a) Protection of structure, sequence and organization (SSO)

i) SSO as copyrightable

A number of judicial decisions have endorsed the view that copyright protection does not limit itself to the mere literal text of a program. A landmark case in this regard was Whelan Assocs., Inc. v Jaslow Dental Laboratory., Inc.[24] The Whelan case, concerned "a dispute between a software consultant and her customer. The consultant prepared a computer program for Jaslow Dental Laboratory ('Jaslow') to manage its inventory and accounts. A written agreement provided that the consultant was to retain 'ownership' of the rights in the program although Jaslow would own one copy of the program. Whelan intended to market the program and make royalty payments to Jaslow. Several years later, Jaslow was licensed to distribute the program, a licence which was subsequently revoked by Whelan. Jaslow claimed that one of its principals, Ran Jaslow, was the owner of the program under the 'work for hire' doctrine, or in the alternative, that he was a co-author of the program due to his explanation of laboratory operations and assistance in the design of some of the program's visual displays. The Court rejected these claims. The interesting part of the case concerned the preparation by Jaslow of an unauthorised version of the program to operate on an IBM-PC. This work involved translating the program from EDL (the original language) to BASIC. This translation could not be done literally because of the difference between the two languages, but the Court found that Jaslow infringed Whelan's copyright by copying 'the manner in which the information flows from one function to another in the program'. In finding an infringement the Court also

---

[23] HR 94-1476, 94th. Cong., 2nd. Sess.57 (1976).

[24] 609 F. Supp. 1307 (E.D. Pa. 1985). aff d, 797 F. 2d 1222 (3d. Cir. 1986), cert. denied U.S. 1301 (1987).

relied on the similarity of the visual output of the two programs" (Radcliffe, 1986, p.44).

In accordance with the Court in Whelan:

> "The expression of the idea in a software computer program is the matter in which the program operates, controls and regulates the computer in receiving, assembling, calculating, retaining, correlating, and producing useful information either on a screen, print-out or by audio communication."

The Whelan judgement has received both enthusiastic support and criticism. While for some it is an evidence of the adaptability of copyright protection to software, for others it is the result of a defective analysis and a source of overprotection for software innovations (Samuelson, 1992. p.12).

Whelan's reasoning was reflected in other court decisions. In SAS Institute, Inc. v S&H Computer Systems Inc.,[25] the court found, in accordance with OTA review (OTA, 1992, p. 70), that S&H infringed the copyright held in a program called SAS 79.5, which was written to run on IBM and IBM compatible computers by converting it to run on Digital computers. The court cited instances of "literal, near literal and organizational copyright", of structural detail and nearly exact duplication of the SAS structure and organization. The court also discussed the idea of merger of idea and expression, stating:

> "... Throughout the preparation of a complicated computer program such as SAS, the author is faced with a virtually endless series of decisions as to how to carry out the assigned task ... At every level, the process is characterized by choice, often made arbitrarily, and only occasionally dictated by necessity. Even in the case of simply statistical calculations, there is room for variation, such as the order in which arithmetic operations are performed ... As the sophistication of the calculation increases, so does the opportunity for variation of expression (OTA, 1992, p.70)."

Finding that the processes of SAS could be expressed in a variety of ways, the Court stated that:

> "... to the extent that similarities between the SAS and the S&H product have existed, they represent unnecessary, intentional duplication of expression."

The SAS case illustrates the application of the so-called "plurality of expression" test. It also showed the difficulties of proof in software lawsuits:

> "Print-outs were destroyed and expert witnesses had to testify based on 'reconstructed' versions of the source code.

---

[25] 605 F. Supp. 816 (M.D. Tenn. 1985).

> Moreover, S&H continued to modify its program during the course of the litigation, frequently eliminating previously detected similarities. At one point during the development of the new program S&H did attempt to avoid direct copying by limiting the access of its programmers to the SAS source code: the programmers were provided with a description of the functions of a module of the SAS program and they were told to duplicate those functions. However, this system frequently failed and the S&H programmers often had direct access to the SAS source code. Consequently, the S&H program contained many instances of verbatim copying of the SAS program." (Radcliffe, 1986, p.43)

Criticisms on the legal appropriateness and implications of the protection of SSO became apparent in other decisions that did not follow or that rejected the Whelan reasoning. Those decisions are summarized in the next sub-section based on OTA, 1992 (p.71-72).

ii) SSO as non-protectible

The Fifth Circuit Court of Appeals did not follow Whelan in Plains Cotton Cooperative Ass'n v. Goodpasture Computer Serv., Inc. Relying on expert testimony, the court found no copying when an allegedly infringing program, designed to run on a personal computer rather than a mainframe computer, was found very similar to the plaintiff's program on the functional specification. Even though the court found the two programs very similar with respect to programming and documentation levels, and found that portions of the design appeared to be direct copies, the court looked to other evidence and found no copying. The court did not adopt the Whelan holding that the structure, sequence, and organization of a computer program is copyrightable. The court held that similarities in the two programs - each of which was designed to perform the same particular task within the agricultural cotton market - were dictated by the "externalities of the market". The record indicated that the market significantly affected the determination of the sequence and organization of cotton marketing software, since both programs attempted to provide the same information to the user. The court did not hold that such patterns could not constitute an idea in the context of computers. Thus, the decision in Plains Cotton narrowed the Whelan decision so that the defendant can show that similarities in structure and organization may be dictated by market factors (externalities) (OTA, 1992, p. 71).

The court in Computer Associates International, Inc. v. Altai, Inc. also rejected the Whelan test of "structure, sequence and organization" to determine similarities in computer programs. Instead, the court applied the "levels of abstraction test" articulated by Learned Hand in Nichols v. Universal Pictures, which, they stated, was the law of the Second Circuit Court of Appeals. The "levels of abstractions test" of Nichols reads:

> "Upon any work ... a great number of patterns of increasing generality will fit equally well, as more and more of the

> incident is left out. The last may perhaps be no more than the most general statement of what the (work) is about and at times might consist only of its title; but there is a point in this series of abstractions where they are no longer protected, since otherwise the (author) could prevent the use of his "ideas" to which, apart from their expression, his property is never extended."

Applying this test, the court found no infringement of Computer Associates' copyright (OTA, 1992, p.72).

To the extent that Computer Associates v. Altai may be interpreted as ruling that a similar structure is not necessarily a copyright infringement, this case may give rise to a new phase in the evolution of software protection in the United States, characterized in this case by a narrowing down of the extent of protection. It illustrates how difficult it continues to be to square computer software protection with the copyright law, and confirms 1985 OTA's views in the sense that policy on software protection "is being made in the courts, virtually on a case by case basis, and the resulting ambiguities satisfy no one" (OTA, 1985, p. 34).

b) Look and feel

Another area of testing of copyright law as applied to software has related to the protection of the "look and feel" of a particular program. The trend in case law is similar to the one described in the preceding paragraphs.

In Broderbund Software, Inc. v. Unison World,Inc.,[26] the court extended the program code copyright to protect both the program display and user interface, holding that the interface idea did not merge with the expression. The decision was taken even in the absence of a valid copyright on the program's output, on grounds that some analysts' judgement was based on an erroneous reasoning (Scholkoff, 1988).

More relevant for the industry have been decisions in Digital Communications Assocs. v Softklone Distributing Corp.,[27] and particularly, in Lotus Development Corp. v. Paperback Software International.[28] In the former the court concluded that the copied elements of the defendant's programs that were nonessential to program operation constituted copyrightable expression. It rejected the defense that idea and expression merged in the "Crosstalk" screen and that the status screen was only an unprotectible "blank form" designed to record the user's choices of parameter values.

---

[26] 648 F. Supp. 1127 (N.D. Cal. 1986).

[27] 659 F. Supp. 449 (N.D. Ga. 1987).

[28] 740 F. Supp. 37 (D. Mass. 1990).

The Lotus decision extended the copyrightability of the non-literal elements of computer programs to menu command structures. The structure, sequence, and organization of the menu command system were all found copyrightable, including the overall structure, the choice of letters, words, or "symbolic tokens" used to represent each command, the structure and order of the command terms in each menu line, the representation of the command terms on the screen, and the long prompts (OTA, 1992, p. 73).

Lotus Development (Lotus) owned the copyright in the Lotus 1-2-3 program. Lotus convinced the Court that the majority of the creative effort and value of 1-2-3 lies in the conceptualization and creation of the user interface. The Court said:

> "I credit the testimony of expert witnesses that the bulk of the creative work is in the conceptualization of a computer program and its user interface is a more difficult intellectual task, requiring greater creativity, originality, and insight, than converting the user interface design into instructions to the machine."

Stephenson developed his own spreadsheet program, between 1982 and 1984. The publisher of his program was Paperback Software. They called this program V-P Planner (VP-P). Because 1-2-3 was so successful, Stephenson and Paperback decided to modify their program to make it 'compatible' with 1-2-3. They created an identical macro sub-language and duplicated almost all of the screen display and command menu structure. In May 1985, Paperback began marketing VP-P as a "worka-like for 1-2-3". The program was actually released in late October, 1985. By January of 1986, Lotus had actual notice of VP-P.

On January 20, 1987, Lotus received a letter from the Copyright Office rejecting a separate registration for the "screen displays" of 1-2-3. Lotus filed suit for copyright infringement. It alleged that Paperback and Stephenson, by copying the macro language, screen displays, and command language, had infringed its copyright.

The main aspects of the referred decision can be summarized as follows:[29]

i) Legal test for copyrightability: the court reduced the basic law of what is copyrightable to a three part test:

> FIRST, in making the determination of "copyrightability", the decision-maker must focus upon alternatives that counsel may suggest, or the court may conceive, along the scale from the

---

[29] This summary is based on the case review published by The Software Law Bulletin, August 1990.

most generalized conception to the most particularized, and choose some formulation, conception or definition, of the "idea" for the purpose of distinguishing between the idea and its expression.

SECOND, the decision-maker must focus upon whether an alleged expression of the idea is limited to elements essential to expression of that idea (or is one of only a few ways of expressing the idea) or instead includes identifiable elements of expression not essential to every expression of that idea.

THIRD, having identified elements of expression not essential to every expression of the idea, the decision-maker must focus on whether those elements are a substantial part of the allegedly copyrightable "work".

This process must be used on a case by case basis. It does not lend itself to clear, bright-line answers.

ii) The 'user interface' is the part of a program which helps the computer system to communicate with the person using it. There are many parts to the user interface: it includes the controls for printers and video displays, the formatting and style of messages, the 'help' functions, and so forth. The 'screen displays' properly include only those aspects on the program actually affecting the video display.

Paperback argued that the 'screen display' was a separate audiovisual work and must be independently copyrighted. The Court felt that Paperback had confused a copyright on the interface aspects of the program with a more limited attempt to protect the appearance of the program on the video screen:

The copyrightable element is not, strictly speaking, the screen display, narrowly understood, but the literal and non-literal elements not only of the display but also of the distinctive way of creating it.

iii) The court held that the

"... particular expression of a menu structure is not essential to the electronic spreadsheet idea, nor does it merge with the somewhat less abstract idea of a menu structure for an electronic spreadsheet. The idea of a menu structure - including the overall structure, the order of commands in each menu line, the choice of letters, words or "symbolic tokens" to represent each command, the presentation of these symbolic tokens on the screen ... the type of menu system used ... and the long prompts - could be expressed in a great many if not literally unlimited number of ways.

The fact that some of these specific commands terms are quite obvious or merge with the idea of such a particular command term does not preclude copyrightability for the command structure taken as a whole ... The Statutory provisions

regarding "compilation", 17 U.S.C. 101, 103, are not essential to this conclusion but do reforce it."

The importance of the described decision for the software industry is indicated by the reaction it caused both in the United States and abroad. For those promoting a more open framework for software development, the decision went too far in curtailing competition through the use of features that become popular with users:

> "The Lotus victory sets a precedent that threatens to stymie the entire software industry. The new lawsuits attempt to extend this monopoly even further. If this ruling is not reversed, either by higher courts or by new legislation, it will become impossible for software startups to compete with companies selling established products, because users will be uninterested in new products for which they must learn a new set of commands." (Stallman, 1991, p. 36)

The continuity of this trend towards enhancing software protection seems, however, uncertain today, as indicated by two recent decisions. In Engineering Dynamics, Inc. v Structural Software Inc. and S. Rao Guntur,[30] the court did not follow the reasoning of Lotus. Citing Plains Cotton, the court held that formats are not copyrightable. Engineering Dynamics claimed the defendants infringed on several of its manuals in the development and marketing of the defendant's product StruCAD. It also claimed the defendants infringed on its copyright in the "user interface", comprised mainly of input and output reports. The court found that the scope of infringed materials included the text, pictures, diagrams, illustrative examples and flow charts depicted in the manuals, but not the input and output formats since the law of the Firth Circuit provides that a user interface in the form of input and output reports is not copyrightable (OTA, 1992, p. 73).

In a long awaited ruling, in Apple v. Microsoft and Hewlett-Packard, the U.S District Court, San Francisco, decided in 1992, that only specific screen elements are copyrightable, and not overall "look and feel".[31] This decision is viewed by some as a turning point for "look and feel" protection, though the future of the matter is still uncertain.

2.2 European and Japanese Law

A provision like the one contained in Article 102 b) of the US Copyright Act is not familiar to European law. However, the EC

---

[30] Civ. Act. Nº 89-1655.

[31] Business Week, July 20, 1992, p. 49.

Directive on the Legal Protection of Computer Programs contains a specific rule drawing up the dividing line between protectible and non-protectible subject matter. Article 1.2 states that:

> "Protection in accordance with this Directive shall apply to the expression in any form of a computer program. Ideas and principles which underlie any element of a computer program, including those which underlie its interfaces, are not protected by copyright under this Directive."

It is doubtful whether this provision, as drafted, adds something to existing copyright law. Unlike earlier drafts, it does not refer to algorithms and programming languages.[32] It also eliminated a confusing text relating to "the specification of interfaces" which "constitutes ideas and principles".

Under Japanese copyright law, protection does not extend to any programming language, rule or algorithm used for making a program (art. 10, para. 3). These concepts are clarified as follows:

> "(i) "Programing language" means letters and other symbols as well as their systems for use as means of expressing a program;
>
> (ii) "Rule" means a special rule on how to use in a particular program a programming mentioned in the preceding item;
>
> (iii) "Algorithm" means methods of combining, in a program, instructions given to a computer."

In connection with the specified exceptions, Prof. Kitagawa argues that "these exceptions are necessary because since a program can only be completed through a variety of technological intermediaries (e.g., flowchart, algorithm) and with the help of programming language or interface information, the idea-expression dichotomy in computer programs does not become very meaningful without identifying which of these matters belongs to the expression of a program" (Kitagawa, 1989, p. 13).

In accordance with another authority:

> "In contrast to the leading cases in the United States, which seem to afford very broad protection to computer programs, the general opinion in Japan is that, just like scientific works, computer programs enjoy rather thinner scope of protection than works such as novels, which are given rather broad protection even to their plots and basis structures." (Tatsumi, p. 16-17).

In line with the above opinion, a case judged by the Tokyo

---

[32] For a criticism on the inclusion of these latter concepts, see Vanderberghe, 1989, p.410.

High Court ruled that "a sequence of operations" contained in a computer program is a "method of solution" non-protectible under Japanese copyright law (Tokyo High Court Order, 20 June 1990, Hanrei Jijo N° 1322, p. 138).

2.3 Summary

This section has shown how difficult it is to establish the frontiers between protectible and non-protectible elements of software. The idea/expression dichotomy provides a general guidance, but it is clearly insufficient to judge specific cases in such a complex technical context. The issues dealt with are not only of juridical interest: the way in which they are solved may have important implications on competition patterns and on the rate of innovation and diffusion. Lack of protection may affect potential investment in developing new products; but over-protection is likely to erect entry barriers which are too high and slow down the rate of generation and diffusion of innovations.

It is certainly difficult to strike a balance between all the interests involved. Recent trends in US case law and the Japanese position seem to indicate, however, that a too broad protection (including elements of internal sequence and structure and "look and feel") has not reached consensus; a more flexible approach is likely to prevail in the 1990's. This view is partially confirmed by developments relating to "reverse engineering" to be considered in the next section.

3. REVERSE ENGINEERING AND SOFTWARE DEVELOPMENT

In view of the preceding discussion, a relevant question for a software producer is to what extent the concepts, ideas and structure of an existing program can be used to independently develop a program which may be attached to or substitute the original one? Is it possible to use the same screen symbols, command structures and other aspects of user interfaces? Is it legitimate to use reverse engineering and other methods to obtain the underlying information of an existing program?

It is, of course, impossible to reply to these questions without taking into consideration a specific legal system and its principles and rules. However, multilateral negotiations on intellectual property are framing international standards on the matter, strongly influenced by developments and approaches in the most advanced countries. In the present international scenario, the room for national legislation to adopt particular solutions in the area of intellectual proper / have also been limited by the use by some developed countries of trade-related sanctions as a means to ensure high standards of protection

Whatever the legal framework, a better knowledge of policy issues and how to safely envisage software development at the firm level is important. This section is intended to provide elements in that respect.

3.1 What is "Reverse Engineering" in the Software Area?

Reverse engineering, or reverse analysis, is the study of a product to understand its functional aspects and underlying ideas. In 1989 a United States District Court defined it in a software copyright infringement case as:

> "... a fair and honest means of ... starting with the known product and working backwards to analyze how the product operates or it was made".[33]

There is a multiplicity of methods of reverse engineering in the software area. They range from reading manuals and other materials to more complex decompilation and disassembly.[34] Based on ECIS,[35] the following methods may be distinguished:

READING MANUALS: The simplest method of reverse analysis is studying manuals that describe a program's operation. This type of analysis is so commonplace that most people would not think of it as "reverse engineering", but it is. The study of a manual does not give the reader the right to republish a similar manual.

TEST RUNS: In this common reverse engineering technique, data are fed into a program, and the output is examined to analyze how the program operates. The data are continually altered, and the resulting output is examined, until a particular process or method of operation is understood.

CONNECTION TESTS: These are used to develop and test communication programs. Messages are sent from a program running on one computer, to a program running on another computer, and the results are analyzed. "Line traces", analyzing signals transmitted over a communication line, can be used to determine the required timing, sequence, and meaning of communication protocols. Failures may be analyzed by memory dumps (discussed below) to correct errors.

STORAGE MEDIA DUMPS: Computer programs organize data according to predefined formats. This technique prints the data directly from a disc or other storage medium, permitting study of the format in which a program records data, so that an engineer can write a new

---

[33] Secure Servs. Technology Inc. v. Time & Space Processing. Inc. 722 F. Supp. 1354, 1361, n. 16 (E.D. Va. 1989).

[34] In the debate on the legal admissibility of reverse engineering, this concept is often limited to decompilation. In turn, decompilation is being used to include disassembly (OTA, 1992, p. 148). These notions, however, should be properly used in order to adequately deal with the legal issues (see also in this regard Kitagawa, 1989).

[35] "Reverse analysis of computer programs", undated document.

program that can read data stored by the first program, such as a word processing program that can read and edit documents written by another word processing program.

MEMORY DUMPS: A memory dump consists of printing or displaying on a screen the portion of a program or data residing temporarily in the main memory of a computer. The memory dump is displayed in machine or "object" code (zeros and ones) and can be understood only by the most expert programmers. This technique is often used when a program stops in mid-stream due to technical difficulties.

REVERSE ASSEMBLY AND REVERSE COMPILATION: Reverse assembly and reverse compilation (or "decompilation") permit analysis similar to the analysis permitted by memory dumps, but do not require that programmers read object code. Reverse assembly usually consists of converting zeros and ones into assembly language, a language perceptible to some skilled programmers. Decompilation - attempting to convert assembly language into an approximation of the original source code - is essentially a misnomer because so much of the original source code that is lost during compilation cannot be restored by decompilation. Decompilation is, thus, better understood as a series of techniques for understanding the logic of a program available only in object code form.

The main legal problems have been posed by the above-mentioned methods of reverse engineering: decompilation and disassembly. One of the reasons for this is that copyright law generally confers on the title-holder the exclusive right to reproduce the copyrighted work, and reverse engineering requires such reproduction in order to be implemented (Soltysinski, 1990, p. 458-459).

As mentioned, the product of decompilation and disassembly would never be identical to the original source program. "At the very least, comments and the names of labels, variables, and procedures would be lost in the assembly or compilation process and could not be recovered. In addition, the structure of the decompiled program would not necessarily be the same as that of the original program, although this would depend on the compiler that had been used. Because of the loss of mnemonics and much of the structure of the program, considerable work is required to understand the decompiled or disassembled program" (OTA, 1992, p. 147).

Decompilation and disassembly are complex, time-consuming and laborious tasks. As noted by the EC Commission in the Explanatory Memorandum accompanying its original proposal for the Software Protection Directive, "although it is technically possible to decompile a program in order to find out information concerning access protocols and interfaces this is a lengthy, costly and inefficient procedure".[36] The process requires considerable skills and while executing it the programmers must supply information not available in the computer executable program. If the purpose of the

---

[36] O.J. Eur. Comm. (N° C 91) 8 (1989).

process is to obtain a "clone" program, once the programmer has completed the analysis and determined the detailed specifications, he has to initiate the software engineering development in order to transform the specifications into a new source code.

Decompilation and disassembly are not conducted lightly (Vinje, 1991, p. 6) nor used routinely in the industry. It is not comparable to "piracy", i.e. the mere reproduction of an existing program by copying of its object code. The Office of Technology Assessment did not find evidence indicating that decompilation is widely used by "pirates" to decompile entire programs and then rearrange the code in an attempt to hide copying (OTA, 1992, p. 148).

3.2 Reasons for Reverse Engineering

Reverse analysis may be employed not only for the development of compatible or substitute products. The main reasons for its use are the following:

Debugging

To "debug" or fix problems in a program distributed in a machine-readable format, an end-user or third-party maintenance provider must be able to disassemble portions of the program, insert "break points" where the program will stop its operation, and examine the computer's memory values to find where the problem occurs.

Systems integration

Systems integration requires extensive reverse analysis to learn how products from several vendors will best work together, with or without additional software.

Interoperable product development

The development of products which may interoperate with existing products is one of the main paths of growth of the informatics industry and of diffusion of the computer technology. In order to create programs that interoperate, it is necessary to know the employed interfaces and protocols. It may also be necessary to know the layout (structure and content) used when the program writes information to a data file on a disk.

Research and development of competitive products

A fourth reason for reverse engineering is to develop a product that would be a substitute and compete with the original one. Incremental innovation is fundamental in the software industry and hence a basic reason to analyze existing programs is to understand underlying ideas and the state of the art in order to improve upon them.

3.3 The Legal Issues in USA, Europe and Japan

As mentioned above, the legal problem of admissibility of reverse engineering is linked to the scope of exclusive rights conferred by copyright, particularly as regards to the reproduction involved in decompilation processes. Legal systems in industrialized countries diverge on the solution to be given to this problem, as indicated by recent legislation and case law which is reviewed below.

a) United States: A changing scenario

US legislation does not address specifically the issue of decompilation.[37] Many computer lawyers have held that since decompiling a program involves making an unauthorized copy of the program, this copy itself is a copyright infringement. It would also constitute the misappropriation of a trade secret and a breach of license agreements usually restricting decompilation (Samuelson, 1992, p. 7). It was generally admitted, however, that a possible statutory defence of unauthorized reproduction was provided by the "fair use" doctrine as contained in Article 107 of the Copyright Act. In accordance with the Committee on the Judiciary of the House of Representatives, "U.S. copyright law does not provide a categorical, statuary, exception to exclusive rights expressly authorizing decompilation or other reverse engineering activities. The copyright liabilities of programers, decompilers and other reverse engineers is determined under the law of fair use, codified in 17 USC 107. The flexibility of the fair use doctrine permits users, proprietors and courts to judge such proposed use or case on its merits. It avoids the rigidity of a specific exception which, as technology changes over time, can become grossly unfair to copyright owners as well as users."[38]

Case law on this issue has contributed to clarify the admissibility of decompilation. In NEC Corp. v Intel Corp,[39] the court did not condemn the disassembling of an Intel microcode

---

[37] In fact, the situation is the same in the rest of the world with the sole exception, at present, of the EC Directive on Computer Programs which is commented below. Amendments to copyright laws to incorporate software issues have been mainly concerned with piracy policies and with the regulation of reproduction, adaptation and, in some case, distribution rights.

[38] Note addressed to the United States Trade Representative on 21.2.90.

[39] 67.434 (N.D. Cal. Feb. 6, 1989).

(8086/88 microprocessor chips) for the purpose of researching and developing a competitive microcode program:

> "... The court found no infringement, basing its holding on the following findings:
>
> 1. No substantial similarity of the works "considered as a whole";
>
> 2. Insufficient evidence that NEC copied important parts of Intel's microcode;
>
> 3. Programming "constraints" accounting for similarities between the two microcodes; and
>
> 4. The limited number of ways in which to express the ideas underlying some of Intel's more basic microroutines.
>
> The findings of the court were particularly well supported through the evidence of "Clean Room" microcode presented to the court. NEC had contended that many of Intel's microsequences were not copyrightable because they were made up of only a few obvious steps and thus lacked the originality necessary for copyright protection... The court looked to Clean Room microcode, developed by a third party, as compelling evidence that the similarities between the NEC microcode and the Intel microcode resulted from constraints. It found that the Clean Room microcode was governed by the same constraints as applied to the NEC microcode, and that copying was not involved" (OTA, 1992, p. 71-72).

A recent case, Sega Enterprises Ltd. v Accolade Inc,[40] has confirmed the interpretation suggested by Nec v. Intel decision in more categoric and precise terms. The court ruled that intermediate copying necessary for disassembly of computer object code is a fair use where it is the only way to gain access to the ideas and functional elements embodied in a copyrighted computer program and where there is a legitimate reason to seek such access. Accolade had made an intermediate copy and disassembly of a Sega video game object code to make compatible video games. The court specifically dealt with the use of such techniques in order to develop a competitive product. It stated that:

> "... an attempt to monopolize the market by making it impossible for others to compete runs counter to the statutory purpose of promoting creative expression and cannot constitute a strong equitable basis for resisting the invocation of the fair use doctrine."

Sega v. Accolade also sided with Computer Assoc. v Altai (see above, section 2) in turning away from the reasoning in Whelan v Dental, holding that because of the hybrid nature of computer

---

[40] CA 9, Nº 92-15655, 10.20.92.

programs, there is no settled standard for distinguishing protected expression from an unprotected idea in a computer copyright case.

b) Europe

As already mentioned, the approval of the EC Directive on the Protection of Computer Programs gave raise to an intense debate on reverse engineering and decompilation issues. Article 5.3 provides that a person having a right to use a copy of a computer program is entitled "to observe, study or test the functioning of the program in order to determine the ideas and principles which underlie any element of the program if he does so while performing any of the acts of loading, displaying, running, transmitting or storing the program which he is entitled to do".

The solution reached on decompilation is contained in Article 6 "Decompilation" (the text of the Directive is reproduced in Annex 3). The Directive permits decompilation when it is "indispensable to obtain the information necessary to achieve the interoperability of an independently created program with other programs".

Article 6 means, first, that such information could not be obtained from other sources, e.g. from written published materials. In practice, information contained in these materials is frequently incomplete or too inaccurate to develop an interoperable product.

Second, that decompilation is only legitimate to achieve "interoperability". Two products are "interoperable" (or "compatible") when they are interchangeable and can function together as part of the same system, despite their being eventually developed independently by different producers.

Third, interoperability need not be related (as proposed in previous drafts of the Directive) to the original decompiled program, but to "other programs". This has been interpreted[41] as allowing for the development not only of programs that attach to the original one but also of programs that compete with or substitute it on the market. Such an interpretation has been confirmed by the EC's own Commission in its communication to the European Parliament of January 1991 and in its Twentieth Annual Competition Law Report:

> "The Commission's concern (regarding the Computer Programs' Directive) was to ensure that a fair balance was maintained between, on the one hand, the protection of the rights attaching to the program and, on the other, the safeguarding of an economic environment that could encourage competition and innovation on the market ... Decompilation is permitted to the extent necessary to ensure the interoperability of an independently created computer program. Such a program may

---

[41] See ECIS, "WIPO memorandum on questions concerning a possible protocol to the Berne Convention. ECIS Comments" (undated document).

> connect to the program subject to decompilation. Alternatively it may compete with the decompiled program and in such cases will not normally connect to it. Article 6 does not however permit decompilation beyond what is necessary to achieve the interoperability of the independently created program. It cannot therefore be used to create a program reproducing parts of a decompiled program having no relevance to the interoperability of the independently created program."[42]

Article 6 of the EC Directive contains, in addition to the mentioned general conditions for decompilation, a number of specifications for decompilation to legally proceed: it should be conducted by a person authorized to use a copy of the program; the necessary information for interoperability should not be previously readily available; the acts should be confined to parts of the program necessary to achieve interoperability.[43]

c) Japan

Like in the United States, there is no specific provision on decompilation in the copyright law. However, based on the distinction between protectible and unprotectible subject matter (see section 1, above), the prevailing legal opinion seems to favour a flexible approach. While pointing out the "gap" existing on this issue in modern copyright law and the various techniques of reverse engineering, Prof. Kitagawa suggests that from a legal point of view reverse engineering may be analyzed only after the type of technique used has been ascertained (Kitagawa, 1989, p. 21).

In accordance with another authority, reverse engineering should be deemed as "fair use" and if the copyright owner "claims infringement because of his or her computer program being reverse engineered, such a claim should be regarded as an abuse of his or her copyright and should be dismissed" (Tatsumi, p. 20). In the view of the quoted author:

> "The essence of computer programs lies in technology, and technology in general progresses step by step and incrementally on the basis of the achievements accomplished by our predecessors. In this respect, there is a need for access to be allowed to the technology contained in computer programs in order to promote development of better software technology. There may also be a need for knowing interface specifications or communication protocols contained in computer programs in

---

[42] EC Commission, CM 60 91 410 Part II, ch. 2.1.4 (g), p. 67-68.

[43] This latter condition might be interpreted as providing a de facto protection to ideas which are unprotectible under copyright principles. The same would apply to Article 6.2.a (see annex and Vinje, 1992, p.7)). Such an interpretation would, however, be contradictory with the expression/idea distinction as dealt with in Article 1.2 of the Directive.

order to develop compatible machines or to realize interconnection between computers. Meeting these needs will promote the advanced of software technology and activate the computer-related information market in a way desirable to the users of computers systems. However, in order to meet these needs, computer programs developed by others may sometimes have to be disassembled or decompiled, or otherwise analyzed and the permissibility of this act of analyzing other computer programs called "the reverse engineering" is disputed and discussed internationally, especially when it accompanies reproduction or adaption of computer programs."
(Tatsumi, p. 18-19)

As expressed, the position in Japan on reverse engineering would not limit its legal admissibility (as suggested also by the Sega V. Accolade case in the United States) to the information necessary for the interoperability of programs like under the EC Directive. Such a method might be employed for gaining access, in principle, to any ideas and functional elements embodied in a protected program.

3.4 Implications for Software Development

Issues considered in section 2 and in the preceding paragraphs are complex and subject to diverging solutions under different laws examined above. However uncertain the situation still is in industrialized countries, in developing countries the scenario is even less defined. Given the relatively low significance of original software development in the latter countries and the weight of imported packaged software therein, it is not surprising that legislation (wherever enacted or amended) and case law has not specifically dealt with "second generation" issues or did so incidentally.

How can software be safely developed in that scenario? The reply to this question should take into account , first, that the software market is largely internationalized and that any significant development is likely to transcend national boundaries. Second, to the extent that the development of new programs may be based on reverse engineering techniques applied to foreign-owned programs, there would be a likelihood of legal conflicts with more powerful and law-skilled program owners. Third, such a reply should be provided at two levels:

- At the policy level, it would be important to establish clear rules that reduce uncertainty on possible developments and create an appropriate environment to that end;

- At the enterprise level, promoting software development techniques that are compatible with the legal framework and able to avoid later troubles and contention.

a) Policy level

At the policy level, a debate seems still necessary on the best ways of reconciling software protection with incremental innovation and improvement on programs. The access to underlying ideas and concepts seems essential for that purpose. Even while recognizing that protection should not be merely limited to literal elements of the expression, it should not go so far as to embrace structural aspects that should remain non-appropriable. Reverse engineering methods should be, therefore, allowed both to develop attachable or substitute products in order to promote competition, provided that the new product is not substantially similar to the original one. In this regard, a clear distinction should be made between the development process, on the one side, and the end product, on the other: if the latter is strikingly similar to the original in its expression, it would be infringing whatever process was used.

b) Enterprise level

As illustrated by the SAS v. S&H case (see section 2, above), an appropriate documentation of the different phases of the development process is essential from a legal perspective. Adequately preserved print-outs of source code may be a critical element for judicial proof in case of alleged infringement. Moreover, in order to be prepared for an idea/expression or reverse engineering dispute, software producers may adopt specific procedures, such as "double blind development, outside development, isolated company design team, non-isolated internal development, etc." (Pilny, 1992, p. 211).

The "clean room" procedure is a well known reverse engineering practice in which "one team of software developers studies the code of a copyrighted program to extract the underlying functionality (ideas). A second team (which has never had access to the copyrighted code) then creates a new program, based on the first team's functional specifications" (OTA, 1990, p. 20).

Given the type of software, the degree of eventual dispute may be different. Thus, in the case of microcode programs, as shown by the NEC v. Intel case, and of operating systems, the multiplicity of expressions is considerably limited by the functional specifications of the program. Another relevant element may be the existence of an economic relationship (e.g. as employee or subcontractor) with the owner of the original program.

4. MAIN CONCLUSIONS

1) International developments on software protection

International developments on software protection, particularly within GATT, clearly point towards the affirmation of

copyright as the main form of software protection.[44] However, the TRIPs Draft Agreement has only dealt with limited aspects of the problem: a) the idea/expression dichotomy is addressed in a general way, leaving the implementation of the principle to national legislations; b) software is treated as a literary work, with its implications on the term of protection and the application of other rules of the Berne Convention; c) rental rights are specifically provided for.

No specific provision is found in said draft dealing with decompilation and appropriability of user interfaces (and "look and feel" aspects), two of the main issues still searching for a solution.

The negotiation of an eventual protocol to the Berne Convention may provide an opportunity for the establishment of new standards on those matters.

2) Protection of ideas and expression

The study made indicates that the underlying ideas and concepts of a program are not deemed protectible under copyright law in general. Recent US case law seems to narrow down previous judicial interpretation extending protection to the structure, sequence and organization of a program. Given the complexity of this issue, divergences are likely to prevail for a long time among different national laws.

3) Reverse engineering

Reverse engineering, including decompilation and disassembly, are admitted as a legitimate means of development of interoperable or competitive products, under differing conditions, according to reviewed national and European laws. Considerable uncertainty exists, however, on the precise limits of permitted and prohibited acts.

The situation regarding this topic is particularly dramatic in developing countries, where neither legislation nor case law have yet addressed the complex legal and technical aspects involved. Those countries should develop legal frameworks conducive for innovation and sound competition in the software industry, and actively participate in the framing of international standards compatible with the use of legitimate techniques of software enhancement and development.

---

[44] Though not dealt with in this paper, the growing acceptance of software patents, at least in the United States, is a trend to be carefully followed.

ANNEX 1

DRAFT AGREEMENT ON TRADE-RELATED ASPECTS OF INTELLECTUAL PROPERTY RIGHTS, INCLUDING TRADE IN COUNTERFEIT GOODS
(Text as of 21.12.91)

PART II: STANDARDS CONCERNING THE AVAILABILITY, SCOPE AND USE OF INTELLECTUAL PROPERTY RIGHTS

SECTION 1: COPYRIGHT AND RELATED RIGHTS

Article 9: Relation to Berne Convention

1. Members shall comply with Articles 1-21 and the Appendix of the Berne Convention (1971). However, Members shall not have rights or obligations under this Agreement in respect of the rights conferred under Article 6bis of that Convention or of the rights derived therefrom.

2. Copyright protection shall extend to expressions and not to ideas, procedures, methods of operation or mathematical concepts as such.

Article 10: Computer Programs and Compilations of Data

1. Computer programs, whether in source or object code, shall be protected as literary works under the Berne Convention (1971).

2. Compilations of data or other material, whether in machine readable or other form, which by reason of the selection or arrangement of their contents constitute intellectual creations shall be protected as such. Such protection, which shall not extend to the data or material itself, shall be without prejudice to any copyright subsisting in the data or material itself.

Article 11: Rental Rights

In respect of at least computer programs and cinematographic works, a Member shall provide authors and their successors in title the right to authorise or to prohibit the commercial rental to the public of originals or copies of their copyright works. A Member shall be excepted from this obligation in respect of cinematographic works unless such rental has led to widespread copying of such works which is materially impairing the exclusive right of reproduction conferred in that Member on authors and their successors in title. In respect of computer programs, this obligation does not apply to rentals where the program itself is not the essential object of the rental.

Article 12: Term of Protection

Whenever the term of protection of a work, other than a photographic work or a work of applied art, is calculated on a basis other than the life of a natural person, such term shall be no less than fifty years from the end of the calendar year of

authorised publication, or, failing such authorised publication within fifty years from the making of the work, fifty years from the end of the calendar year of making.

Article 13: Limitations and Exceptions

Members shall confine limitations or exceptions to exclusive rights to certain special cases which do not conflict with a normal exploitation of the work and do not unreasonably prejudice the legitimate interests of the right holder.

Article 14: Protection of Performers, Producers of Phonograms (Sound Recordings) and Broadcasting Organizations

1. In respect of a fixation of their performance on a phonogram, performers shall have the possibility of preventing the following acts when undertaken without their authorisation: the fixation of their unfixed performance and the reproduction of such fixation. Performers shall also have the possibility of preventing the following acts when undertaken without their authorisation: the broadcasting by wireless means and the communication to the public of their live performance.

2. Producers of phonograms shall enjoy the right to authorise or prohibit the direct or indirect reproduction of their phonograms.

3. Broadcasting organisations shall have the right to prohibit the following acts when undertaken without their authorisation: the fixation, the reproduction of fixations, and the rebroadcasting by wireless means of broadcasts, as well as the communication to the public of television broadcasts of the same. Where Members do not grant such rights to broadcasting organisations, they shall provide owners of copyright in the subject matter of broadcasts with the possibility of preventing the above acts, subject to the provisions of the Berne Convention (1971).

4. The provisions of Article 11 in respect of computer programs shall apply mutatis mutandis to producers of phonograms and any other right holders in phonograms as determined in domestic law. If, on the date of adoption of the Final Act Establishing the Results of the Uruguay Round of Multilateral Trade Negotiations a Member has in force a system of equitable remuneration of right holders in respect of the rental of phonograms, it may maintain such system provided that the commercial rental of phonograms is not giving rise to the material impairment of the exclusive rights of reproduction of right holders.

5. The term of the protection available under this Agreement to performers and producers of phonograms shall last at least until the end of a period of fifty years computed from the end of the calendar year in which the fixation was made or the performance took place. The term of protection granted pursuant to paragraph 3 above shall last for at least twenty years from the end of the calendar year in which the broadcast took place.

6. Any Member may, in relation to the rights conferred under paragraphs 1-3 above, provide for conditions, limitations, exceptions and reservations to the extent permitted by the Rome Convention. However, the provisions of Article 18 of the Berne Convention (1971) shall also apply, mutatis mutandis, to the rights of performers and producers of phonograms in phonograms.

Preparatory Memorandum by the WIPO Committee of Experts on the Berne Convention with Regard to the Protection of Literary and Artistic Works, November 1991

COMPUTER PROGRAMS

19. The WIPO "Model Provisions on the Protection of Computer Software" published in 1978 define computer programs as follows: "A 'computer program' is a set of instructions expressed in words, codes, schemes or in any other form, which is capable, when incorporated in a machine-readable medium, of causing a 'computer'—an electronic or similar device having information-processing capabilities--to perform or achieve a particular task or result." This definition is more than ten years old but it still seems to be usable since it reflects appropriately the essential elements of the notion of computer programs.

20. The first question that should be discussed is whether it may be considered an obligation under the Berne Convention to protect computer programs as a category of literary and artistic works.

21. Article 2(1) of the Berne Convention provides that "[t]he expression 'literary and artistic works' shall include every production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression." This general description of the meaning of "literary and artistic works" is followed by a non-exhaustive list of such works.

22. It is not stated explicitly in Article 2(1) of the Berne Convention, but the records of the various diplomatic conferences adopting and revising the Berne Convention--and, in respect of collections, also the text (Article 2(5)) of the Convention itself--indicate that the "productions" considered works are those which constitute original intellectual creations.

23. The questions of the intellectual property protection of computer programs were first discussed at the international level during the preparation of the Model Provisions on the Protection of Computer Software referred to in paragraph 19, above.

24. At that time, the relative newness of computer technology and the scarcity of relevant legislation and case law resulted in considerable uncertainty in this field. The 1978 Model Provisions offered certain minimum provisions that constituted a *sui generis* system. At the same time, the commentary to the said Model Provisions stressed that they should not be understood as necessarily requiring adoption of a separate *sui generis* law, and that they could be implemented by a copyright law.

25. Since then, the trends towards copyright protection has prevailed over *sui generis* protection throughout the world.

26. Notwithstanding this prevailing trend, it cannot be said that the professional circles uniformly believe that copyright is the best kind of protection of computer programs, and that granting copyright protection is required by Berne Convention.

27. The following main doubts were raised concerning copyright protection of computer programs: (a) the purpose of computer programs is to cause a computer--that is, a machine--to perform or achieve a particular task or result, which is alien to the notion of literary and artistic works; (b) although computer programs in source code form can be perceived by human beings, that is not the case in respect of computer programs in object code form; (c) computer programs are frequently made of sub-routine elements, and such programs may not be considered original; (d) copyright laws cannot be

applied directly for the protection of computer programs; specific provisions are necessary; (e) the 50 year post mortem auctoris minimum term of protection under the Berne Convention is too long, since computer programs usually become outdated in a much shorter time; (f) copyright does not protect algorithms, which are considered the most fundamental creative elements of computer programs.

28. When copyright protection for computer programs was granted in various countries, either by legislation or by court decisions, the following answers were given to the doubts referred to in the preceding paragraph: (a) computer programs are basically writings, and, under Article 2(1) of the Berne Convention, the purpose for which writings are created is irrelevant from the viewpoint of their qualifying as literary works; not only works of literature proper, but also scientific writings and writings with a purely utilitarian or commercial aim should be protected as literary works, if they are original intellectual creations; (b) computer programs in object code form share the copyright status of other literary and artistic works stored in computer systems in machine-readable form; they can be retreived—"decompiled"—into a form in which they are available to human beings; if it were true that works in machine-readable form—from which they can be retrieved and made available in such a way--were not protected, that would be the end of copyright protection because, with the quick development of computer technology, nearly all categories of works can be included in computer systems in such a way; storage of works in a computer system must be considered reproduction; (c) with the exception of a few simple programs, there is sufficient room for creativity in making computer programs; unless an unreasonably high originality test is applied, nearly all computer programs would pass such a test; (d) the need for specific provisions does not mean that the protection of a certain category of works would be alien to copyright; the Berne Convention and national copyright laws also include specific provisions in respect of various other categories of literary and artistic works; (e) the alleged problem of too long term of protection is of an academic nature; there are a number of other categories of literary and artistic works which may become obsolete within a much shorter period than 50 years; the latter should be considered nothing other than an upper limit; (f) it is quite appropriate that copyright does not protect algorithms—as it does not protect any idea, procedure, process, method of operation, concept, principle or discovery, in general—but only the concrete expressions thereof; this is precisely why copyright can offer appropriate protection for computer programs without creating unreasonable obstacles to independent creation of such programs.

29. The reasons discussed in the preceding paragraph seem convincing, and have also been justified by the actual application of copyright protection of computer programs in a number of countries.

30. It is proposed that the possible Protocol

(a) should provide that countries party to the Protocol are obliged to grant copyright protection to computer programs and that the protection must be the same (subject to certain exceptions specified below) as the Berne Convention provides for literary and artistic works, and

(b) should indicate that the notion of computer programs comprises both operation system programs and application programs, whether in source code form or in object code form.

31. One of the consequences of such provisions in the proposed Protocol would be that all provisions of the Berne Convention which do not deal with special kinds of works, as for example cinematographic works, would apply also to computer programs. Such provisions (mentioned in the order in which they appear in the Berne Convention) would, in particular (since the following list is not exhaustive), include the following:

(i) the protection of computer programs would "operate for the benefit of the author and his successors in title" (Article 2(7)); if one should allow that the protection operate in favor of someone else, for example the person who directs the creation of the computer program, the Protocol should so state; this solution would be analogous to the solution of Article 14bis of the Berne Convention concerning cinematographic works;

(ii) the criteria of eligibility for protection under the Berne Convention (Article 3(1) and (2)) would apply also to computer programs; if, however, one would adopt the solution indicated in item (i), in fine, above, the Protocol should parallel Article 4(a) in respect of computer programs;

(iii) the definition of publication (Article 3(3)) should also apply to computer programs; however, since most computer programs are not made available in a number of copies sufficient "to satisfy the reasonable requirements of the public," most computer programs would remain unpublished, and the rules concerning unpublished works should apply to them;

(iv) the requirement of national treatment and the minimum rights ("rights specially granted by the Convention") (Article 5(1)) would apply to computer programs; any present national law not respecting those minima would have to be modified;

(v) the enjoyment and the exercise of these rights (that is, national treatment and the rights specially granted by the Convention) could not be subject to any formality (Article 5(2)) even in the case of computer programs;

(vi) the prohibition of reciprocity (Article 5(1)) would (subject to the possibility of "comparison of terms" under Article 7(8) and the unimportant and so far never applied exception provided for in Article 6), apply to computer programs; consequently, those national laws that today allow reciprocity would have to be changed;

(vii) moral rights (Article 6bis) would apply to authors of computer programs, unless the solution referred to in item (i), in fine, is adopted; but even if that solution is not adopted, moral rights could be claimed in practice rarely since in most cases the authors of computer programs are unidentifiable;

(viii) the minimum term of protection generally applicable (Article 7(1), (3) and (5) to (8) and 7bis) would apply also to computer programs; if one would adopt the solution indicated in item (i), in fine, above, the Protocol should provide for a term of protection for computer programs that are similar to the one provided for in Article 7(2) for cinematographic works, namely, for 50 years from their publication, or, in the absence of publication, for 50 years from their making; consequently, those national laws that today protect computer programs for less than 50 years (from the death of the author, from first publication or from first making available to the public, as the case may be) would have to be changed;

(ix) the exclusive right of authorizing alterations (Article 12) would apply to computer programs; the transformation of computer programs from one computer "language" into another should be considered to be covered by this right rather than by the right of translation (Article 8) in view of the fact that the notion of translation under the Convention was and is intended to cover real, that is, human languages, and that the use of the word "language" in the field of computers is merely a symbolic designation;

(x) the provisions on the right to enforce protected rights (Article 15(1), (2) and (4)) would apply to computer programs;

(xi) the provisions on seizure (Article 16) would apply to computer programs.

32. Of the provisions on the minima under the Berne Convention, there seems to be one, namely Article 9(2)--free reproduction in certain special cases--concerning which, when applied to computer programs, the views of professional circles significantly differ, and concerning which governments having legislated or planning to legislate in this field do not seem to agree.

33. The question is which are the special cases where free reproduction of computer programs does not conflict with normal exploitation and does not unreasonably prejudice the legitimate interests of copyright owners and, thus, where such reproduction can be allowed.

34. Although differences in views still exist, there seems to be growing agreement concerning the following points: (a) taking into account the purpose and value of computer programs, free copying for private purposes--except for cases that are covered by points (b) and (c), below--should not be allowed; (b) free copying by lawful owners, that is, persons who have acquired ownership of copies of (not of the copyright in) computer programs should be allowed in certain circumstances; (c) free decompilation of computer programs (see paragraph 37, below) should also be allowed under certain conditions.

35. As regards paragraph 34(b), above, it is obvious that copying should be allowed if it is indispensable for the use of a program in conjunction with a machine for the purpose, and to the extent of use, for which the program has been lawfully obtained. Furthermore, it also seems justified to allow making a "back-up copy" for archival purposes, as a security measure, for cases where the replacement of the program may become necessary.

36. In addition to clarifying the extent to which a lawful owner of a computer program may make a copy, it also seems necessary to make it clear that the right of adaptation under Article 12 of the Berne Convention does not include the right to prevent an adaptation that is indispensable for using the computer program in conjunction with a machine for the purpose, and to the extent of use, for which the program has been lawfully obtained.

37. Decompilation of computer programs means reproduction and adaptation ("translation") of computer programs into a form in which the coding and structure of the program can be examined and analyzed. It seems that such decompilation by lawful owners of computer programs should be allowed, since such decompilation would not conflict with any normal exploitation of the program and would not cause any unreasonable prejudice to the legitimate interests of copyright owners, in cases where decompilation is needed to obtain information necessary to achieve interoperability of independently created programs with the original programs concerned (from which it follows that decompilation must not be allowed if such information is readily available from other sources), in respect of those parts of the original program concerned that are necessary to achieve interoperability. However, to avoid any conflict and prejudice referred to above, the information thus obtained should not be used for the development, production or distribution of a program substantially similar in its expression to the original program, or for any other act infringing copyright.

38. It is proposed that the possible Protocol should provide

(a) that, without the authorization of the owner of the copyright in it, it is, subject to (b) and (c) below, not permitted to reproduce a computer program for private purposes and

(b) that it is a matter for national legislation to permit the lawful owner of a copy of a computer program to make, without the authorization of the owner of the copyright in the computer program, another copy or an adaptation of such a program, provided that such a copy or adaptation is

(i) indispensable for using the computer program in conjunction with a machine for the purpose, and to the extent of use, for which the program has been lawfully obtained; or

(ii) for archival purposes and, if necessary (in the event that the original copy of the program is lost, destroyed or rendered unusable), for the replacement of the copy lawfully obtained; provided that such a copy or

adaptation may not be used for any purposes other than the ones mentioned above and must be destroyed when the continued possession of copies or adaptations of the computer program ceases to be lawful;

(c) it is also a matter for national legislation to permit the lawful owner of a copy of the computer program to decompile, without the authorization of the owner of the copyright in the computer program, the program into a form in which its coding and structure can be examined, provided that

(i) such decompilation should only be allowed in cases where the information necessary to achieve interoperability of other independently created computer programs with the original program concerned is not readily available from other sources, and only in respect of those parts of the original program concerned that are necessary to achieve interoperability;

(ii) the information obtained through such decompilation may only be used to achieve interoperability of an idependently created computer program, and may not be used for making a program substantially similar in its expression to the original program, or for any other act infringing copyright.

39. These three provisions, that is, those proposed in points (a), (b) and (c) of the preceding paragraph, would be compatible with the Berne Convention, because they would simply clarify, for certain situations, the meaning of Article 9(2) of the Berne Convention, and the provision under (a) also because it gives a "greater protection" (Article 19) to computer programs than to other types of works.

No L 122/42 Official Journal of the European Communities 17. 5. 91

II

(Acts whose publication is not obligatory)

COUNCIL

COUNCIL DIRECTIVE

of 14 May 1991

on the legal protection of computer programs

(91/250/EEC)

THE COUNCIL OF THE EUROPEAN COMMUNITIES,

Having regard to the Treaty establishing the European Economic Community and in particular Article 100a thereof,

Having regard to the proposal from the Commission (1),

In cooperation with the European Parliament (2),

Having regard to the opinion of the Economic and Social Committee (3),

Whereas computer programs are at present not clearly protected in all Member States by existing legislation and such protection, where it exists, has different attributes;

Whereas the development of computer programs requires the investment of considerable human, technical and financial resources while computer programs can be copied at a fraction of the cost needed to develop them independently;

Whereas computer programs are playing an increasingly important role in a broad range of industries and computer program technology can accordingly be considered as being of fundamental importance for the Community's industrial development;

Whereas certain differences in the legal protection of computer programs offered by the laws of the Member States have direct and negative effects on the functioning of the common market as regards computer programs and such differences could well become greater as Member States introduce new legislation on this subject;

Whereas existing differences having such effects need to be removed and new ones prevented from arising, while differences not adversely affecting the functioning of the common market to a substantial degree need not be removed or prevented from arising;

Whereas the Community's legal framework on the protection of computer programs can accordingly in the first instance be limited to establishing that Member States should accord protection to computer programs under copyright law as literary works and, further, to establishing who and what should be protected, the exclusive rights on which protected persons should be able to rely in order to authorize or prohibit certain acts and for how long the protection should apply;

Whereas, for the purpose of this Directive, the term 'computer program' shall include programs in any form, including those which are incorporated into hardware; whereas this term also includes preparatory design work leading to the development of a computer program provided that the nature of the preparatory work is such that a computer program can result from it at a later stage;

Whereas, in respect of the criteria to be applied in determining whether or not a computer program is an original work, no tests as to the qualitative or aesthetic merits of the program should be applied;

(1) OJ No C 91, 12. 4. 1989, p. 4; and OJ No C 320, 20. 12. 1990, p. 22.
(2) OJ No C 231, 17. 9. 1990, p. 78; and Decision of 17 April 1991, not published in the Official Journal).
(3) OJ No C 329, 30. 12. 1989, p. 4.

Whereas the Community is fully committed to the promotion of international standardization;

Whereas the function of a computer program is to communicate and work together with other components of a computer system and with users and, for this purpose, a logical and, where appropriate, physical interconnection and interaction is required to permit all elements of software and hardware to work with other software and hardware and with users in all the ways in which they are intended to function;

Whereas the parts of the program which provide for such interconnection and interaction between elements of software and hardware are generally known as 'interfaces';

Whereas this functional interconnection and interaction is generally known as 'interoperability'; whereas such interoperability can be defined as the ability to exchange information and mutually to use the information which has been exchanged;

Whereas, for the avoidance of doubt, it has to be made clear that only the expression of a computer program is protected and that ideas and principles which underlie any element of a program, including those which underlie its interfaces, are not protected by copyright under this Directive;

Whereas, in accordance with this principle of copyright, to the extent that logic, algorithms and programming languages comprise ideas and principles, those ideas and principles are not protected under this Directive;

Whereas, in accordance with the legislation and jurisprudence of the Member States and the international copyright conventions, the expression of those ideas and principles is to be protected by copyright;

Whereas, for the purposes of this Directive, the term 'rental' means the making available for use, for a limited period of time and for profit-making purposes, of a computer program or a copy thereof; whereas this term does not include public lending, which, accordingly, remains outside the scope of this Directive;

Whereas the exclusive rights of the author to prevent the unauthorized reproduction of his work have to be subject to a limited exception in the case of a computer program to allow the reproduction technically necessary for the use of that program by the lawful acquirer;

Whereas this means that the acts of loading and running necessary for the use of a copy of a program which has been lawfully acquired, and the act of correction of its errors, may not be prohibited by contract; whereas, in the absence of specific contractual provisions, including when a copy of the program has been sold, any other act necessary for the use of the copy of a program may be performed in accordance with its intended purpose by a lawful acquirer of that copy;

Whereas a person having a right to use a computer program should not be prevented from performing acts necessary to observe, study or test the functioning of the program, provided that these acts do not infringe the copyright in the program;

Whereas the unauthorized reproduction, translation, adaptation or transformation of the form of the code in which a copy of a computer program has been made available constitutes an infringement of the exclusive rights of the author;

Whereas, nevertheless, circumstances may exist when such a reproduction of the code and translation of its form within the meaning of Article 4 (a) and (b) are indispensable to obtain the necessary information to achieve the interoperability of an independently created program with other programs;

Whereas it has therefore to be considered that in these limited circumstances only, performance of the acts of reproduction and translation by or on behalf of a person having a right to use a copy of the program is legitimate and compatible with fair practice and must therefore be deemed not to require the authorization of the rightholder;

Whereas an objective of this exception is to make it possible to connect all components of a computer system, including those of different manufacturers, so that they can work together;

Whereas such an exception to the author's exclusive rights may not be used in a way which prejudices the legitimate interests of the rightholder or which conflicts with a normal exploitation of the program;

Whereas, in order to remain in accordance with the provisions of the Berne Convention for the Protection of Literary and Artistic Works, the term of protection should be the life of the author and fifty years from the first of January of the year following the year of his death or, in the case of an anonymous or pseudonymous work, 50 years from the first of January of the year following the year in which the work is first published;

Whereas protection of computer programs under copyright laws should be without prejudice to the application, in appropriate cases, of other forms of protection; whereas, however, any contractual provisions contrary to Article 6 or to the exceptions provided for in Article 5 (2) and (3) should be null and void;

Whereas the provisions of this Directive are without prejudice to the application of the competition rules under Articles 85 and 86 of the Treaty if a dominant supplier refuses to make information available which is necessary for interoperability as defined in this Directive

Whereas the provisions of this Directive should be without prejudice to specific requirements of Community law already enacted in respect of the publication of interfaces in the telecommunications sector or Council Decisions relating to standardization in the field of information technology and telecommunication;

Whereas this Directive does not affect derogations provided for under national legislation in accordance with the Berne Convention on points not covered by this Directive,

HAS ADOPTED THIS DIRECTIVE:

Article 1

Object of protection

1. In accordance with the provisions of this Directive, Member States shall protect computer programs, by copyright, as literary works within the meaning of the Berne Convention for the Protection of Literary and Artistic Works. For the purposes of this Directive, the term 'computer programs' shall include their preparatory design material.

2. Protection in accordance with this Directive shall apply to the expression in any form of a computer program. Ideas and principles which underlie any element of a computer program, including those which underlie its interfaces, are not protected by copyright under this Directive.

3. A computer program shall be protected if it is original in the sense that it is the author's own intellectual creation. No other criteria shall be applied to determine its eligibility for protection.

Article 2

Authorship of computer programs

1. The author of a computer program shall be the natural person or group of natural persons who has created the program or, where the legislation of the Member State permits, the legal person designated as the rightholder by that legislation. Where collective works are recognized by the legislation of a Member State, the person considered by the legislation of the Member State to have created the work shall be deemed to be its author.

2. In respect of a computer program created by a group of natural persons jointly, the exclusive rights shall be owned jointly.

3. Where a computer program is created by an employee in the execution of his duties or following the instructions given by his employer, the employer exclusively shall be entitled to exercise all economic rights in the program so created, unless otherwise provided by contract.

Article 3

Beneficiaries of protection

Protection shall be granted to all natural or legal persons eligible under national copyright legislation as applied to literary works.

Article 4

Restricted Acts

Subject to the provisions of Articles 5 and 6, the exclusive rights of the rightholder within the meaning of Article 2, shall include the right to do or to authorize:

(a) the permanent or temporary reproduction of a computer program by any means and in any form, in part or in whole. Insofar as loading, displaying, running, transmision or storage of the computer program necessitate such reproduction, such acts shall be subject to authorization by the rightholder;

(b) the translation, adaptation, arrangement and any other alteration of a computer program and the reproduction of the results thereof, without prejudice to the rights of the person who alters the program;

(c) any form of distribution to the public, including the rental, of the original computer program or of copies thereof. The first sale in the Community of a copy of a program by the rightholder or with his consent shall exhaust the distribution right within the Community of that copy, with the exception of the right to control further rental of the program or a copy thereof.

Article 5

Exceptions to the restricted acts

1. In the absence of specific contractual provisions, the acts referred to in Article 4 (a) and (b) shall not require authorization by the rightholder where they are necessary for the use of the computer program by the lawful acquirer in accordance with its intended purpose, including for error correction.

2. The making of a back-up copy by a person having a right to use the computer program may not be prevented by contract insofar as it is necessary for that use.

3. The person having a right to use a copy of a computer program shall be entitled, without the authorization of the rightholder, to observe, study or test the functioning of the program in order to determine the ideas and principles which underlie any element of the program if he does so while performing any of the acts of loading, displaying, running, transmitting or storing the program which he is entitled to do.

Article 6

Decompilation

1. The authorization of the rightholder shall not be required where reproduction of the code and translation of its form within the meaning of Article 4 (a) and (b) are indispensable to obtain the information necessary to achieve the interoperability of an independently created computer program with other programs, provided that the following conditions are met:

(a) these acts are performed by the licensee or by another person having a right to use a copy of a program, or on their behalf by a person authorized to to so;

(b) the information necessary to achieve interoperability has not previously been readily available to the persons referred to in subparagraph (a); and

(c) these acts are confined to the parts of the original program which are necessary to achieve interoperability.

2. The provisions of paragraph 1 shall not permit the information obtained through its application:

(a) to be used for goals other than to achieve the interoperability of the independently created computer program;

(b) to be given to others, except when necessary for the interoperability of the independently created computer program; or

(c) to be used for the development, production or marketing of a computer program substantially similar in its expression, or for any other act which infringes copyright.

3. In accordance with the provisions of the Berne Convention for the protection of Literary and Artistic Works, the provisions of this Article may not be interpreted in such a way as to allow its application to be used in a manner which unreasonably prejudices the right holder's legitimate interests or conflicts with a normal exploitation of the computer program.

Article 7

Special measures of protection

1. Without prejudice to the provisions of Articles 4, 5 and 6, Member States shall provide, in accordance with their national legislation, appropriate remedies against a person committing any of the acts listed in subparagraphs (a), (b) and (c) below:

(a) any act of putting into circulation a copy of a computer program knowing, or having reason to believe, that it is an infringing copy;

(b) the possession, for commercial purposes, of a copy of a computer program knowing, or having reason to believe, that it is an infringing copy;

(c) any act of putting into circulation, or the possession for commercial purposes of, any means the sole intended purpose of which is to facilitate the unauthorized removal or circumvention of any technical device which may have been applied to protect a computer program.

2. Any infringing copy of a computer program shall be liable to seizure in accordance with the legislation of the Member State concerned.

3. Member States may provide for the seizure of any means referred to in paragraph 1 (c).

Article 8

Term of protection

1. Protection shall be granted for the life of the author and for fifty years after his death or after the death of the last surviving author; where the computer program is an anonymous or pseudonymous work, or where a legal person is designated as the author by national legislation in accordance with Article 2 (1), the term of protection shall be fifty years from the time that the computer program is first lawfully made available to the public. The term of protection shall be deemed to begin on the first of January of the year following the abovementioned events.

2. Member States which already have a term of protection longer than that provided for in paragraph 1 are allowed to maintain their present term until such time as the term of protection for copyright works is harmonized by Community law in a more general way.

Article 9

Continued application of other legal provisions

1. The provisions of this Directive shall be without prejudice to any other legal provisions such as those concerning patent rights, trade-marks, unfair competition, trade secrets, protection of semi-conductor products or the law of contract. Any contractual provisions contrary to Article 6 or to the exceptions provided for in Article 5 (2) and (3) shall be null and void.

2. The provisions of this Directive shall apply also to programs created before 1 January 1993 without prejudice to any acts concluded and rights acquired before that date.

Article 10

Final provisions

1. Member States shall bring into force the laws, regulations and administrative provisions necessary to comply with this Directive before 1 January 1993.

When Member States adopt these measures, the latter shall contain a reference to this Directive or shall be accompanied by such reference on the occasion of their official publication. The methods of making such a reference shall be laid down by the Member States.

2. Member States shall communicate to the Commission the provisions of national law which they adopt in the field governed by this Directive.

Article 11

This Directive is addressed to the Member States.

Done at Brussels, 14 May 1991.

For the Council

The President

J. F. POOS

REFERENCES

Business Week. (1989). "Editorials: Don't use copyright to shackle software", May 29.

Correa, Carlos. (1990a). "The legal protection of software", Informatica e Diritto, Anno XVI, gannaio-aprile.

Correa, Carlos. (1990b). "Software industry: An opportunity for Latin America", World Development, Vol. 18, N° 11.

EC Commission. (1988). Green paper on copyright and the challenge of technology, "Copyright issues requiring immediate action", Brussels.

Geller, Paul. (1991). International Copyright Law and Practice, vol. 1. Matthew Bender.

Kitagawa, Zentaro. (1989). "Legal protection of computer programs in Japan. One aspect of technology law", Regional Forum on the Impact of Emerging Technologies on the Law of Intellectual Property, WIPO, Seoul.

Mody, Ashoka. (1989). "New environment for intellectual property", World Bank, Washington D.C.

OTA. (1985). "Information Technology R&D: Critical trends and issues", Washington D.C.

OTA. (1986). "Intellectual property rights in an age of electronics and information", Washington D.C.

OTA. (1990). "Computer Software and Intellectual Property", Background Paper, Washington D.C.

OTA. (1992). "Finding a balance: Computer software, intellectual property and the challenge of technological change", Washington D.C.

Peterson, G. and Makay, C. (1992). "The US Software Rental Amendments Act", Copyright World, N°22, May-June.

Pilny, Karl. (1992). "Legal aspects of interfaces and reverse engineering - Protection in Germany, the United States and Japan", IIC, vol. 23, N° 2.

Radcliffe, Mark. (1986). "Recent US developments in copyright law related to computer software", European Intellectal Property Review, N° 2, February.

Samuelson, Pamela. (1992). "Adapting intellectual property law to new technologies: A case study on computer programs", Conference on the Global Dimensions of Intellectual Property Rights in Science and Technology, National Research Council, Washington D.C., January 8-9.

Schware, Robert. (1992). "Software industry entry strategies for developing countries: A walking on two legs", World Development, vol. 20, Nº 2.

Sholkoff, Jack. (1988). "Breaking the mold: Forging a new and comprehensive standard of protection for computer software", Computer Law Journal, vol. VII.

Soltysinski, Stanislaw. (1990). "Legal protection for computer programs, public access to information and freedom of competitive research and development activities", Rutgers Computer and Technology Law Journal, vol. 16, Nº 2.

Stallman, Richard. (1991). "Programmers and users protest against Lotus lawsuits", Managing Intellectual Property, Issue 6, March.

Tatsumi, Naohiko. (Undated). "Copyright law and protection of computer programs", Osaka International Training Center, JICA.

Uchtenhagen, Ulrich. (1990). "The GATT negotiations concerning copyright and intellectual property protection", IIC, vol. 21, Nº 6.

Vanderberghe, Guy. (1989). "Copyright protection of computer programs: An unsatisfactory proposal for a Directive", European Intellectual Property Review, Nº 11.

Vinje, Thomas. (1991). "The development of interoperable products under the EC Software Directive", The Computer Lawyer, vol. 8, Nº 11.

Von Lewinski, Silke. (1991). "Rental right, lending right and certain neighbouring rights: The EC Commission's proposal for Council Directive", EIPR, Nº 4.

WIPO (World Intellectual Property Organization). (1988). Background reading material on intellectual property, Geneva.

WIPO. (1992a). "Documento preparatorio para la primera sesión del Comité de Expertos sobre un eventual Protocolo relativo al Convenio de Berna para la protección de las Obras Literarias y Artísticas", Derecho de Autor, Año IV, Nº 2.

WIPO. (1992b). "Comité de Expertos sobre un eventual Protocolo relativo al Convenio de Berna para la Protección de las Obras Literarias y Artísticas, Derecho de Autor, Año IV, Nº 3.