

Computerprogramme sind keine Sprachwerke Teil 2

Michael König

2.

*Maschinenprogramme –
Elektronische Steuerungs-
instrumente*

*Die Programmierung von
EPROM's*

*Computerprogramme und ROM-
Chips*

*Der Quellcode –
Bauplan des Maschinenprogramms*

Aus den vorstehenden Ausführungen ist zu entnehmen, daß ein grundlegender Unterschied zwischen Maschinenprogrammen und Quellcodes besteht.

Maschinenprogramme sind offensichtlich keine immaterielle oder geistige Güter, sondern handfeste elektronische Steuerungsinstrumente. Zur Verdeutlichung wird noch einmal auf die bereits oben angesprochene Möglichkeit, Maschinenprogramme als Zusammenschaltung elektronischer Bauelemente zu realisieren, rekurriert: Ein EPROM besteht nur aus einer Matrix entsprechend miteinander verschalteter elektronischer Funktionseinheiten, nämlich logischer Gatter in OR-, NOR- und NAND-Ausführung. Die „Programmierung“ des EPROMs, also die physische Herstellung des Maschinenprogramms, besteht in der Auflösung entsprechender Verbindungen, so daß sich der „Inhalt“ des EPROMs als eine Folge von „1“en und „0“en – oder auch „H“ und „L“ – darstellen läßt. Zugleich kann man das „programmierte“ EPROM, also das Maschinenprogramm, auch durch eine Zusammenschaltung der entsprechenden elektronischen Bauteile in diskreter, nicht-miniaturisierter Form nachbauen und ersetzen, z.B. durch entsprechende Digital-ICs der 74-TTL-Familie. Unerheblich ist, daß dieser Nachbau mehrere hundertmal größer und teurer als das „programmierte“ EPROM³¹.

Als Folge dieser offensichtlichen Gleichwertigkeit ergibt sich, daß natürlich auch diese Zusammenschaltung elektronischer Bauelemente das betreffende Maschinenprogramm ist. Außerdem läßt dies die Absurdität der Behauptung Kullmarins offenbar werden, nach der „sich die Computerprogramme auf bzw. in den ROM-Chips befinden und nicht das ROM-Chip selbst sind“ und „die Anweisungen, die an die CPU bzw. den Mikroprozessor gegeben werden, ... von den Objektprogrammen kommen“³²: Bei aller Mühe wird man hier kein zwischen, über oder in den ICs verstecktes Maschinenprogramm finden, denn das Maschinenprogramm ist die gesamte Schaltung. Kulmann unterliegt mit vielen anderen dem Irrtum, das Maschinenprogramm als ein von der Verkörperung zu unterscheidendes „geistiges Werk“ anzusehen – ein Irrtum, der letztendlich darauf beruht, daß der technische Charakter aller Computerprogramme als Steuerungsmittel nicht genügend berücksichtigt wird. Nun ist auch für technische Laien zwingend einsichtig, daß das Maschinenprogramm auch durch eine grafische Wiedergabe der Zusammenschaltung der erforderlichen elektronischen Bauteile bzw. der einzelnen logischen Gatter dargestellt werden kann. Diese Darstellung ist indes ein elektronischer Schaltplan und als solcher unstrittig eine Darstellung technischer Art gem. § 2 I Nr.7 UrhG. Es bleibt daher das Geheimnis der h.M. – und von Clapes, Lynch und Steinberg –, wie in dem technischen Steuerungsmittel „Maschinenprogramm“ ein Sprachwerk oder überhaupt ein Werk iSd. Urheberrechts gesehen werden kann.

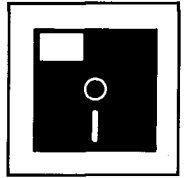
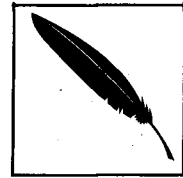
Der Quellcode hingegen ist der Bauplan, die Konstruktionszeichnung für das herzustellende Maschinenprogramm. Elemente des Quellcodes sind bislang Symbole, die häufig der englischen Sprache entnommen wurden. Allein aus der Verwendung dieser Symbole zu folgern, Quellcodes seien Sprachwerke, ist schlicht nicht nachvollziehbar. Es gibt nämlich bereits Programmiersysteme, die auf diese Symbole verzichten. In diesen Programmiersystemen besteht der Quellcode lediglich aus einer Grafik, in der die einzelnen Programmfunktionen durch grafische Symbole dargestellt und mittels unterschiedlichen Linien verbunden sind³³. Es ist bezeichnend, daß man sich emotional dagegen wehrt, diese Programmiersysteme als Programmiersprache zu bezeichnen, obwohl sie funktional gleichwertig sind und ebenfalls zur Herstellung von Maschinenprogrammen dienen und diese produzieren. Eine unterschiedliche Beurteilung beider Arten von Quellcodes ist jedoch nicht zu rechtfertigen. Beide Arten haben sowohl innerhalb der EDV als auch gegenüber dem Programmierer dieselben Funktionen. Sie

Dr. Michael König ist Rechtsanwalt in
Frankfurt.

³¹ vgl. König C.I.2.c)bb)(5)

³² Kullmann 26

³³ Dies ist das von dem Polen Tomasz Pietrzykowski entwickelte Programmiersystem „PROGRAPH“: Klein, Das Programmier-Programm des Dr. P., FAZ 205/90 v. 4.9.90, T6



*Sprachwerk oder
Darstellung technischer Art?*

sind in jeder Hinsicht völlig gleichwertig. Entweder sind auch die offensichtlich grafischen Quellcodes Sprachwerke oder aber herkömmliche Quellcodes sind wie die neuen grafischen Quellcodes zu bewerten – nämlich richtigerweise als Darstellungen technischer Art. Berücksichtigt man, daß auch bei herkömmlichen Quellcodes jedes „sprachliche“ Symbol durch ein „nichtsprachliches“ Symbol ersetzt werden kann und – bei geeigneten Assemblern, Compilern und Interpretern – doch dieselben Ergebnisse und Erfolge erreicht werden, so ist die Entscheidung vorgegeben: Quellcodes sind als Baupläne und Konstruktionszeichnungen der Maschinenprogramme Darstellungen technischer Art.

Hiergegen spricht schließlich auch nicht das oft verwendete Argument, Computerprogramme – und Quellcodes – seien „Anweisungen an den menschlichen Geist“³⁴ und dienten als Lektüre³⁵. Es ist erstaunlich, daß sich dieses Argument so lange gehalten hat, denn es ist absolut untauglich: Mit dieser Argumentation – Programme müßten wegen des Erfordernisses der Änderung etc. von Menschen gelesen und verstanden werden³⁶ – läßt sich jedwede Konstruktionszeichnung, jeder Schaltplan und sogar die danach hergestellten Produkte und Maschinen als Sprachwerke bzw. als Anweisungen an den menschlichen Geist verstehen. Die Vertreter dieser Auffassung mögen sich einmal fragen, was ein Elektroniker unternimmt, wenn er ein elektronisches Gerät zu reparieren hat, dessen Schaltplan er nicht besitzt oder nicht erhalten kann. Die Antwort ist einfach: Der Elektroniker analysiert selbstverständlich die elektronische Schaltung und rekonstruiert aus dieser den bzw. einen Schaltplan – in der Terminologie der Vertreter der besagten Auffassung „liest“ er also die Schaltung. Nur weil ein Fachmann aus jedem technischen Produkt über kurz oder lang dessen Aufbau, Funktion und Wirkungsweise entnehmen kann, stellt dieses doch kein Sprachwerk dar! Das Produkt ist noch nicht einmal als Darstellung technischer Art – wie der zugrundeliegende oder rekonstruierte Schalt-/Bauplan bzw. die Konstruktionszeichnung u.ä. – zu bewerten, sondern ist urheberrechtlich die freie Realisierung dieses Plans und damit offensichtlich kein Werk im urheberrechtlichen Sinne.

„Erschwerend“ und von den Vertretern der besagten Auffassung selbstverständlich völlig ignoriert ist zu berücksichtigen, daß sich herkömmliche Schaltpläne, Konstruktionszeichnungen usw. noch weitaus mehr an den Menschen richten als ein Quellcode: Im Gegensatz zu der automatischen und maschinellen Herstellung eines Maschinenprogramms aus dem Quellcode durch Assembler, Compiler und Interpreter dienen herkömmliche Schaltpläne, Konstruktionszeichnungen usw. der Realisierung durch den Menschen! Wenn man überhaupt von „Anweisungen an den menschlichen Geist“ sprechen will, dann doch offensichtlich nur bei den herkömmlichen Schaltplänen und Konstruktionszeichnungen, denn diese teilen dem Menschen mit, welche Teile er auf welche Art zusammenzubauen hat. Quellcodes „wenden“ sich hingegen in ihren eigentlichen Funktion ausschließlich an den Rechner – hier von „Anweisungen“ zu reden ist völlig abwegig, denn dann müßte man auch z.B. den komplexen, motorgetriebenen Programmwahlschalter älterer mechanischer Waschautomaten als eine solche „Anweisung“ an eben die übrige Waschmaschine ansehen. Hier erübrigt sich jedes weitere Argument.

Listings lesen?

*Anweisung an den menschlichen
Geist*

3.

Vollständig abwegig ist indes die Behauptung von Clapes, Lynch und Steinberg, Quellprogramme seien „literarische Werke“³⁷. Es soll hier nicht auf die Fragwürdigkeit des Arguments eingegangen werden, die Geschäftstätigkeit des Programmierers habe viel mit der eines Buchautors gemeinsam. Solche Parallelen lassen sich zu vielen Berufen ziehen – läßt man einmal die suggestiv verwendeten Begriffe wie „Autor“ und „schreiben“ weg, so stellt man fest, daß dieselbe Kreativität und Aktivität auch bei beratenden Ingenieuren, Erfindern und auch Entwicklern elektronischer Schaltungen anzutreffen ist.

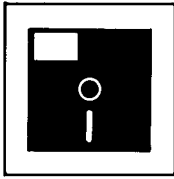
*Abwegig: Quellprogramme als
literarische Werke*

³⁴ z.B. Braun, Rechtsschutz für Rechenprogramme, BB 71, 1343, 1346; Junker Computerrecht: Gewerblicher Rechtsschutz, Mängelhaftung, Arbeitsrecht, JZ 89, 316, 318; Köhler R., Der urheberrechtliche Schutz der Rechenprogramme, 41; Möhring, Die Schutzfähigkeit von Programmen für Datenverarbeitungsanlagen, GRUR 67, 269, 273ff; Skaupy, Zur Schutzfähigkeit von elektronischen Rechenprogrammen, BB 67, 945, 946; Ulmer, Der Urheberrechtsschutz wissenschaftlicher Werke unter besonderer Berücksichtigung der Programme elektronischer Datenverarbeitungsanlagen, 11; Wittmer, Der Schutz von Computersoftware – Urheberrecht oder Sonderrecht?, 94ff

³⁵ Clapes/Lynch/Steinberg jur-pc 89, 230f

³⁶ Clapes/Lynch/Steinberg aaO; Haberstumpf GRUR 82, 142, 148; Preuß 145

³⁷ Clapes/Lynch/Steinberg jur-pc 89, 218, 221ff



Niemand – nicht einmal Clapes, Lynch und Steinberg – werden behaupten wollen, daß der von einem Schaltungsentwickler aufgezeichnete Schaltplan oder sonst ein beliebiger Bauplan ein literarisches Werk sei. Somit gelangt man mit der oben dargelegten und begründeten Auffassung zwangsläufig zu dem Schluß, daß Quellprogramme ebensowenig Werke der Literatur wie Werke der Musik sind. Selbst wenn man diese Auffassung nicht teilen wollte, so würde man doch zugeben müssen, daß Quellcodes – was immer sie auch sein mögen – in jedem Fall doch nur und ausschließlich einen technischen Sachverhalt beschreiben. Sie können also niemals der Literatur, sondern lediglich der Technik oder – im Einzelfall – der Wissenschaft zugeordnet werden. Für das deutsche Urheberrecht – nur dieses ist Gegenstand dieses Beitrags – ist diese Konsequenz offenbar und keiner weiteren Diskussion zugänglich.

IV.

Gestaltungsmöglichkeiten bei der Quellcode-Erstellung

Der Hersteller von Quellcodes ist weitaus unfreier in seinen Gestaltungsmöglichkeiten, als dies auf den Autor von Sprachwerken und sogar den Urheber anderer Darstellungen technischer Art zutrifft. Der Programmierer hat nur eine sehr begrenzte Anzahl von Symbolen („Anweisungen“) zur Verfügung, die er zur Erreichen des angestrebten Ziel einsetzen und kombinieren kann. Auch in der Kombination ist er nicht frei, sondern muß unter allen Umständen die entsprechenden Regeln beachten – ansonsten funktioniert das Programm nicht oder der Compiler bzw. Assembler kann hieraus schon kein Maschinenprogramm herstellen. Dies ist unmittelbar vergleichbar mit dem Entwickler einer elektronischen Schaltung: Wenn dieser den Steuerungsstrom nicht an die Basis, sondern an den Kollektor des Transistors führt, so mag dies zwar „kreativ“ sein, führt jedoch günstigstenfalls zum Nichtfunktionieren der Schaltung, wenn nicht sogar zur Zerstörung des Transistors.

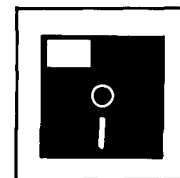
Vollständige Bindung an vorgegebene Symbole

Der Programmierer ist sogar noch nicht einmal in der Gestaltung so frei wie der Entwickler einer elektronischen Schaltung. Dieser kann seinem Schaltplan wenigstens eine individuelle Form geben, indem er die Bauteile bewußt nicht optimal – im Sinne einer optimalen Erfassbarkeit und Verständlichkeit der Schaltung und deren Funktionsweise –, sondern „kreativ“ und nach künstlerischen Gesichtspunkten anordnet. Er kann sogar innerhalb gewisser Grenzen die verwendeten Schaltungssymbole abändern, also den Widerstand trapezförmig oder den Transistor rechteckig oder ellipsoid zeichnen. Der Programmierer hingegen ist nicht nur vollständig an die vorgegebenen Symbole wie „print“ oder „mov“ gebunden – würde er stattdessen „drucke“ oder „move“ verwenden, so könnte dies von dem Compiler bzw. Assembler nicht verarbeitet werden. Sogar die die Anordnung der einzelnen Symbole richtet sich im wesentlichen nach programmtechnischen – also inhaltlichen – Vorgaben. So muß die Definierung einer Variablen vor deren Verwendung erfolgen. Auch können die in einer „for ... next“-Schleife zu erfolgenden Operationen und Funktionen nicht irgendwo außerhalb angeordnet werden. Dem läßt sich nicht entgegenhalten, daß der Programmierer die betreffenden Programmteile irgendwo im Programmlisting positionieren und durch entsprechende Sprung-Anweisungen – „goto“, „do“, „jmp“ u.ä. – dafür sorgen kann, daß die betreffenden Funktionen beim Programmlauf doch ausgeführt werden. Hierbei würde nämlich übersehen werden, daß dies gerade keine rein gestalterische Maßnahme, sondern eine inhaltliche Änderung, also ein anderes Programm, darstellt. Nicht nur sind die danach erzeugten Maschinenprogramme völlig verschieden – das durch diese Sprünge „zerstückelte“ Programm wird aller Wahrscheinlichkeit nach auch weniger leistungsfähig, nämlich zumindest langsamer, sein. Nur am Rande sei erwähnt, daß diese Vorgehensweise auch gegen sämtliche Prinzipien des Software-Engineering verstoßen würde – Quellcodes sollen strukturiert, klar und übersichtlich sein.

Wieviele Wege führen zu einem Programm?

Die in diesem Zusammenhang oft vorgebrachte Behauptung, es gäbe grundsätzlich unendlich viele Möglichkeiten, ein Programm zu entwickeln³⁸, ist zumindest irreführend,

³⁸ BGH I ZR 52/83 v. 9.5.85 NJW 86, 192, 196; Brutschke, Urheberrecht und EDV, 98f; Köhler R. 51ff, 76, 97; Kolle, Der Rechtsschutz von Computerprogrammen aus nationaler und internationaler Sicht, GRUR 74, 7, 8f; ders. GRUR 82, 443, 453; Möhring GRUR 67, 269, 275f; Sieber, Der urheberrechtliche Schutz von Computerprogrammen, BB 83, 977, 980; Ulmer 17; Troller, Der urheberrechtliche Schutz von Inhalt und Form der Computerprogramme, CR 87, 352, 354f; Ulmer/Kolle GRUR Int. 82, 489, 494f, 497; Wittmer 99ff, 139f.



weil sie in diesem Kontext suggeriert, diese vielen Möglichkeiten seien gestalterischer und nicht inhaltlicher Art. Zu berücksichtigen ist nämlich, daß nach dem deutschen Urheberrecht bei Darstellungen technischer Art ein Schutz inhaltlicher Elemente nicht möglich ist und diese auch nicht zur Begründung der Urheberrechtsschutzfähigkeit – der persönlichen geistigen Schöpfung gem. § 2 II UrhG – herangezogen werden dürfen³⁹. Tatsächlich jedoch sind all diese verschiedenen Möglichkeiten, ein Programm zu entwickeln, technisch unterschiedliche Realisierungen – sie sind nicht lediglich gestalterische Möglichkeiten, die u.U. den Anforderungen des § 2 II UrhG genügen würden. Demzufolge ist das zur Einleitung des Beitrags von Clapes, Lynch und Steinberg gebrachte Zitat

„How many different ways are there to produce a program ...?“ – „An infinite number in principle, and in practice dozens, hundreds.“ – „So it is comparable to the theoretically infinite number of ways of writing Hamlet?“ – „I believe so.“

irreführend und unzutreffend. Mit derselben Berechtigung könnte man – hinsichtlich elektronischer Schaltungen –

„How many different ways are there to produce a circuit' ...?“ – „An infinite number in principle, and in practice dozens, hundreds.“ – „So it is comparable to the theoretically infinite number of ways of writing Hamlet?“ – „I believe so.“

formulieren, denn beide Male werden lediglich inhaltliche Realisierungsmöglichkeiten angesprochen, die jedoch – wie mehrfach erwähnt – nicht vom deutschen Urheberrecht erfaßt werden.

Eine Ausnahme hiervon gilt lediglich für die in Quellcodes verwendeten Variablenbezeichnungen, Meldungen/Ausgabertexte und selbstverständlich die den Quellcode erläuternden Kommentare. Nur in dieser Hinsicht könnten Quellcodes als Sprachwerke aufgefaßt werden – auch wenn eine Darstellung technischer Art nicht durch Beschriftungen zu einem Sprachwerk wird⁴⁰ – und u.U. sogar den Anforderungen des § 2 II UrhG genügen. Bei Variablenbezeichnungen wird dies zwar regelmäßig nicht der Fall sein; auch Meldungen/Ausgabertexte geben wenig Gelegenheit zur Individualität, da sie sich – wie die Beschriftung der Bedienungselemente technischer Geräte – dem überragenden Erfordernis der optimalen Verständlichkeit unterordnen müssen. In der Kommentierung hingegen kann der Programmierer seinem „literarischen“ Schaffensdrang freien Lauf lassen. Festzuhalten ist indes, daß diese mit dem dem Programminhalt, dessen Schutz das eigentliche Anliegen der Software-Hersteller ist, nichts zu tun haben.

Hieraus zu folgern, daß Maschinenprogramme eben doch Sprachwerke seien, wäre allerdings zu voreilig. Bekanntlich gehen Variablenbezeichnungen als sprachliche Symbole und Kommentare nicht in das Maschinenprogramm ein⁴¹. Es verbleiben somit die Meldungen/Ausgabertexte, die tatsächlich in dem Maschinenprogramm wiedergefunden werden können⁴². Sie sind jedoch vergleichbar mit der Beschriftung der Frontplatten und Bedienungselementen beliebiger technischer Geräte – die hierdurch selbstverständlich auch nicht zum Werk iSd. Urheberrechts oder gar einem Sprachwerk werden⁴³.

V.

Als Ergebnis könnte man nun festhalten, daß Quellcodes Darstellungen technischer Art sind, deren urheberrechtlicher Schutz allenfalls durch die Kommentierungen und Erläuterungen, keinesfalls jedoch durch inhaltliche Elemente begründet werden kann. Maschinenprogramme hingegen zählen noch nicht einmal zu den Werken iSd. des Urheberrechts, denn sie sind ausschließlich technische Steuerungsmittel der Rechner. Sie entsprechen beliebigen Produkten, die nach einem Schalt- oder Bauplan oder einer Konstruktionszeichnung hergestellt werden – diese Funktion hat der Quellcode. Maschinenprogramme sind daher urheberrechtlich nicht geschützt.

Dieses Ergebnis würde sich zwar in die Systematik des Urheberrechts nahtlos und widerspruchsfrei einfügen und den Gesetzgeber veranlassen, den längst fälligen und

Dazu Revontuli, jur.pc 90/731

Sprachwerke:
Variablennamen, Meldungen,
Kommentare

Kein Urheberrechtsschutz für
inhaltliche Elemente des
Quellcodes ...

... und Maschinenprogramme

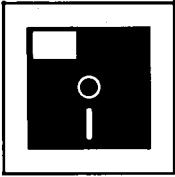
³⁹ s. Fn 13 = jur.pc 05/1991, S. 1099, 1101

⁴⁰ Loewenheim in Schrickler Rdnr. 122 zu § 2

⁴¹ vgl. König GRUR 89, 559, 561, 564

⁴² Eine beliebte, wenn auch sehr primitive Methode des „Nachahmens“ oder „Clonens“ – tatsächlich ist dies jedoch nichts anderes als schlichtes Kopieren – von Computerprogrammen ist daher auch, in dem Maschinenprogramm unter Zuhilfenahme entsprechender Utilities oder – bei MS-DOS – einfach DEBUG diese Texte abzuändern oder zu übersetzen.

⁴³ vgl. König C.II.3.b)bb)(1)(b) a.E., C.II.3.b)bb)(2)(b)



Die Wertung des Gesetzgebers in der UrhG-Novelle

§ 2 I Nr. 1 UrhG: Erweiterung der Werkarten

Das Erfordernis der persönlichen geistigen Schöpfung

sinnvoll begrenzten Sonderschutz für Computerprogramme zu etablieren. Es widerspricht jedoch bedauerlicherweise der letztlich eindeutigen Wertung des Gesetzgebers, die in der Novellierung des Urheberrechtsgesetzes 1985 zum Ausdruck gekommen ist⁴⁴. Der Gesetzgeber wollte Computerprogramme durch deren Aufnahme in § 2 I Nr. 1 UrhG sowie deren Ausnahme von der Zulässigkeit der Herstellung privaten Zwecken dienender Vervielfältigungen in § 53 IV 2 UrhG Computerprogrammen urheberrechtlich schützen. Dieser Absicht widersprechen jedoch die oben als richtig festgestellten Ergebnisse. Eine Reduzierung der gesetzgeberischen Absicht allein auf Quellprogramme, die dann als neue Werkart gelten würden – „... Sprachwerke, wie Schriftwerke und Reden, sowie Programme für die Datenverarbeitung“ –, ist nicht möglich, denn das absolute Kopierverbot sollte gezielt der Bildung eines Marktes für Raubkopien entgegenwirken⁴⁵. Eine Beschränkung auf Quellprogramme würde diesem Zweck widersprechen, denn regelmäßig werden nur Maschinenprogramme vertrieben und sind daher kopiergefährdet. Trotz der lediglich klarstellenden Funktion der Aufnahme von Computerprogrammen in § 2 I Nr. 1 UrhG⁴⁶ ist es zur Wahrung der gesetzgeberischen Intention daher unabdingbar, der Neuregelung nicht nur klarstellende, sondern bei neueren Erkenntnissen – wie oben – darüberhinaus auch neuschaffende Funktion zuzuweisen⁴⁷.

Die Ergänzung von § 2 I Nr. 1 UrhG zu „Sprachwerke, ..., sowie Programme für Datenverarbeitung“ stellt demnach eine Erweiterung der Werkarten um eben Computerprogramme dar, zu denen auch die ansonsten nicht schützbaeren Maschinenprogramme zählen⁴⁸. Entgegen der obigen Feststellungen sind wegen dieser Novellierung des Urheberrechtsgesetzes auch Quellcodes nicht als Darstellungen technischer Art, sondern als Computerprogramme zu schützen, da Quellcodes wegen der ausdrücklichen Bezugnahme in § 53 IV 2 UrhG auf Computerprogramme gem. § 2 I Ziff. 1 UrhG nur auf diese Art vor privater Vervielfältigung geschützt werden können.

Da Maschinenprogramme, wie oben ausgeführt, niemals die erforderliche Schöpfungshöhe aufweisen können, widersprüche eine Beibehaltung der Anforderungen aus § 2 II UrhG dem erklärten gesetzgeberischen Ziel. Andererseits können jedoch Maschinenprogramme in Ermangelung schutzfähiger Elemente nicht umfanglicher als Quellcodes geschützt sein. Zur Wahrung der gesetzgeberischen Intention muß man daher hinsichtlich der Anforderungen an eine persönliche geistige Schöpfung (nur) bei Maschinenprogrammen von einer Neuregelung ausgehen. Dies muß sich indes darauf beschränken, den Schutz der Quellprogramme auf den der hiernach hergestellten Maschinenprogramme einfach zu übertragen, ohne daß hier weiter das Vorliegen schöpferischer Elemente zu prüfen wäre⁴⁹. Die Intention des Gesetzgebers ist jedoch bei jeder Anwendung urheberrechtlicher Vorschriften auf Maschinenprogramme zu beachten, denn nur diese rechtfertigt den Verzicht auf das Erfordernis der persönlichen geistigen Schöpfung bei Maschinenprogrammen: Es sollte der Bildung eines Marktes für Raubkopien entgegengewirkt werden⁵⁰.

Da der Schutz des Maschinenprogramms von dem des Quellcodes abhängt, dieser entsprechend der obigen Darstellung jedoch in Ermangelung schutzfähiger gestalterischer Elemente praktisch nicht existiert, erfordert eine angemessene Berücksichtigung der Absicht des Gesetzgebers eine weitere Korrektur: Zwar sollten die Voraussetzungen der Schutzfähigkeit nicht angetastet werden⁵¹; andererseits sollte die Aufnahme von

⁴⁴ Das Ergebnis der nachfolgenden Überlegungen wurde den an anderer Stelle erfolgten Ausführungen ohne weitere Begründung zugrundegelegt: König GRUR 89, 559, 562, 563 Fußnote 69.

⁴⁵ BT-Drucksache 10/3360, 19; vgl. auch Bericht der Bundesregierung BT-Drucksache 11/4929, 43

⁴⁶ BT-Drucksache 10/3360, 18; BT-Drucksache 11/4929, 41

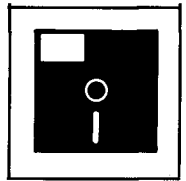
⁴⁷ so auch Preuß 150; a.A. Schulze G., Können Objektprogramme urheberrechtlich geschützt sein? – Kritische Bemerkung zu König, GRUR 89, 559ff – GRUR 90, 103, 104

⁴⁸ so auch Preuß aaO

⁴⁹ Die Richtigkeit dieser Bewertung ergibt auch daraus, daß bis heute hinsichtlich des Schutzes von Computerprogrammen zwischen Quellcodes und Maschinenprogrammen nicht differenziert, sondern von deren Gleichbehandlung ausgegangen wird: vgl. BT-Drucksache 11/4929, 41ff; i.E. ebenso Kindermann, Urheberrechtsschutz von Computerprogrammen, CR 89, 880, 881. In anderem Zusammenhang versteht Troller das Maschinenprogramm als funktionelles Element des Rechners, das als unmittelbare Objektivation des Geisteswerks Computerprogramm wie der Quellcode geschützt sei: Troller CR 87, 352, 353f. Zum Erreichen des mit der Novellierung verfolgten Zwecks ist daher das ausdrückliche Festhalten an den Schutzvoraussetzungen des § 2 II UrhG – BT-Drucksache 10/3360, 18 – auf Quellcodes zu beschränken.

⁵⁰ zu dieser Konsequenz speziell hinsichtlich der Frage, ob die Programm Benutzung eine urheberrechtlich relevante Werkbenutzung darstellt, König C.II.3.c)dd(1)(b)(aa), (dd), C.II.3.c)dd(2)(a), (c)

⁵¹ BT-Drucksache 10/3360, 18; vgl. auch Schulze G. GRUR 90, 103, 104



Computerprogramme in § 2 I Nr.1 UrhG hinsichtlich der damaligen Rechtslage nur klarstellende Funktion besitzen⁵². Da dem Gesetzgeber aber lediglich der Tenor des „Inkassoprogramm“-Urteils des BGH vom 9.5.85⁵³, nicht jedoch dessen Begründung bekannt war⁵⁴, so kann dies durchaus dafür sprechen, daß der Gesetzgeber die bereits zuvor beschrittenen Wege zur Begründung der Schöpfungshöhe durch Heranziehung auch inhaltlicher Elemente⁵⁵ durch die Novellierung gebilligt hat⁵⁶.

Es führt somit kein Weg daran vorbei, in Respektierung des Willens des Gesetzgebers die Neuregelung in § 2 I Nr.1 UrhG dahingehend auszulegen, daß entgegen der oben als richtig erkannten Feststellungen auch inhaltliche Elemente zur Begründung der Schöpfungshöhe von Quellcodes herangezogen werden dürfen. Dies hat freilich unter Beachtung der durch den BGH in dem Urteil „Inkassoprogramm“ aufgestellten Anforderungen zu geschehen, denn der Gesetzgeber hat die Rechtsprechung des BGH akzeptiert und davon abgesehen, durch entsprechende gesetzliche Regelungen eine weitergehende Berücksichtigung des Inhalts von Computerprogrammen zu erreichen⁵⁷. Allerdings hält sich der hierdurch angerichtete Schaden in der Urheberrechtssystematik in Grenzen, denn dank der hohen Anforderungen des BGH dürfte kaum ein Programm urheberrechtlichen Schutz genießen^{58 59}. Es bedarf indes keiner weiteren Erörterung, um zu erkennen, daß dieser Mißbrauch des Urheberrechts dennoch ausgesprochen unerwünscht ist und schleunigst durch eine entsprechende Rückgängigmachung der Urheberrechtsnovelle von 1985 in Verbindung mit der Schaffung eines sinnvoll beschränkten Sonderschutzes für Computerprogramme beseitigt werden sollte.

Auch inhaltliche Elemente zur Begründung der Schöpfungshöhe

*Forderung:
Rückgängigmachung der
Urheberrechtsnovelle*

Zusammenfassung

Die sachgerechte Berücksichtigung technischer Grundlagen hat gezeigt, daß an sich Maschinenprogramme keine Werke iSd. Urheberrechtsgesetzes sind und Quellcodes lediglich als Darstellungen technischer Art geschützt sein können, wobei inhaltliche Elemente zum Erreichen der erforderlichen Schöpfungshöhe weder beitragen dürfen noch können. Die mit der Novellierung des Urheberrechtsgesetzes 1985 verfolgte Absicht des Gesetzgebers, der Bildung eines Marktes für Raubkopien entgegenzuwirken, zwingt jedoch zu der urheberrechtssystemwidrigen Konsequenz, Maschinenprogramme wie Quellcodes und diese entsprechend den Anforderungen des BGH aus dem „Inkassoprogramm“-Urteil zu schützen. Diese Konzession an den Willen des Gesetzgebers vermag jedoch die Richtigkeit der Erkenntnis nicht zu ändern, daß nämlich Computerprogramme – sei es Maschinenprogramme oder Quellcodes – alles andere als „literarische“ Sprachwerke sind.

⁵² BT-Drucksache 10/3360, 18; BT-Drucksache 11/4929, 41

⁵³ BGH I ZR 52/83 v. 9.5.85 NJW 86, 192

⁵⁴ Kindermann CR 89, 880, 881

⁵⁵ OLG Koblenz 6 U 294/80 v.13.8.81 BB 83, 992; OLG Nürnberg 3 U 652/83 v. 8.5.84 GRUR 84, 736; OLG Frankfurt 14 U 188/81 v. 6.11.84 GRUR 85, 1049; Brutschke 98f, 104ff; Köhler R. 51ff; Kolle GRUR 82, 443, 453; Ulmer/Kolle GRUR Int.82, 489, 494f, 497; vgl. die ausführliche Darstellung bei Preuß 160ff

⁵⁶ Hierfür spricht ferner, daß die Rechtsprechung des BGH akzeptiert und davon abgesehen wird, durch entsprechende gesetzliche Regelungen eine weitergehende Berücksichtigung des Inhalts zu erreichen: BT-Drucksache 11/4929, 43.

⁵⁷ vgl. BT-Drucksache 11/4929, 43

⁵⁸ zur Kritik an diesen Anforderungen z.B. Betten, Der Rechtsschutz von Computer-Software (Einführung in das Thema), Mitt. 86, 10; Brandi-Dohrn, Der Schutz von Computersoftware in Rechtsprechung und Praxis, GRUR 87, 1; Heymann, Softwareschutz und EG-Richtlinienentwurf, CR 90, 9, 11; Lehmann, Erwidern, CR 90, 94f; Röttinger, Abkehr vom Urheberrechtsschutz für Computersoftware?, IuR 86, 12; Schmidt/Knorr, Die Werkhöhe bei Software, IuR 86, 7; Schulze G., Urheberrechtsschutz von Computerprogrammen – geklärte Rechtsfrage oder bloße Illusion?, GRUR 85, 997; Zahrt IuR 86, 23

⁵⁹ aus dem Urteil des OLG Hamm 4 U 196/86 v. 27.4.89, CR 89, 592 wird dies sowie die Ungeeignetheit des Urheberrechts für den Schutz von Computerprogrammen deutlich; kritisch hierzu die Anmerkungen von Bücken CR 89, 798 sowie Schulze G. CR 89, 799.