

# Interoperabilität von Software

## Art. 6 der Computerprogramm-Richtlinie aus heutiger Sicht

Prof. Dr. **Andreas Wiebe**, LL.M., Universität Göttingen.

**Abstract:** This article reviews Article 6 of the Software Directive and discusses the need for a revision. Beyond clarification of the scope of the very limited provision on reverse engineering, it seems that the introduction of the clause into copyright was unfortunate. The indirect protection of ideas by prohibiting reverse engineering is foreign to the copyright concept. Permitting reverse engineering altogether

would promote research and development and further other goals like ICT security. Innovation would not be retarded, which is the reason why US trade secret law permits reverse engineering based also on economic arguments. The notions of compatibility Article 6 tries to address are better dealt with by Competition Law, which was demonstrated by the Microsoft Decision of the European Court in 2007.

**Keywords:** Reverse engineering, decompilation, interoperability, copyright limitation, research, protection of ideas, know-how, interfaces, competition law

© 2011 Andreas Wiebe

Everybody may disseminate this article by electronic means and make it available for download under the terms and conditions of the Digital Peer Publishing Licence (DPPL). A copy of the license text may be obtained at <http://nbn-resolving.de/urn:nbn:de:0009-dppl-v3-en8>.

Recommended citation: Andreas Wiebe, Interoperabilität von Software: Art. 6 der Computerprogramm-Richtlinie aus heutiger Sicht, 2 (2011) JIPITEC 89, para. 1.

### A. Einführung

- 1 Es gab keine Vorschrift der RL, die im Gesetzgebungsverfahren so umstritten war wie Art. 6. Im Prozess des Erlasses der RL hat es einen erbitterten Kampf der Lobbygruppen um Reichweite und Formulierung der Regelung gegeben. Herausgekommen ist die längste, detaillierteste und unübersichtlichste Vorschrift der RL. In seltsamem Kontrast dazu steht die Tatsache, dass man heute in den einschlägigen Kommentierungen keine einzige Entscheidung zu dieser Vorschrift findet, die im deutschen Recht in § 69e UrhG umgesetzt wurde.
- 2 Das könnte man nun darauf zurückführen, dass die Vorschrift so detaillierte Regelungen enthält, dass ihre Klarheit jeglichen Rechtsstreit verhindert hat.<sup>1</sup> Man darf allerdings nicht die Detailliertheit einer Regelung mit Klarheit verwechseln, was sich gerade an dem Beispiel von § 69e UrhG zeigt. Eine weitere Erklärung könnte darin liegen, dass die Rechtsdurchsetzung weitgehend im Kartellrecht stattfindet. IBM wurde bereits in den achtziger Jahren in kartellrechtlichen Verfahren gezwungen, Schnittstelleninformationen offen zu legen und dies hatte natürlich auch Auswirkungen auf das Verhalten anderer Her-

steller. Schließlich bleibt, wenn ein Hersteller wirklich ein Verletzungsverfahren anstrengen sollte, auch ein gravierendes Nachweisproblem, da der Nachweis einer Übernahme von geschützten Informationen wiederum ein – nicht erlaubtes – Dekompilieren des angeblich übernehmenden Programms erforderlich machen würde.<sup>2</sup>

- 3 Ich möchte im Folgenden das zugrundeliegende Problem näher eingrenzen, die Lösung der Richtlinie darstellen und die Reichweite der Erlaubnis in § 69e näher untersuchen. Sodann möchte ich der Frage nachgehen, inwieweit hier das Kartellrecht hinreichende Lösungsmöglichkeiten bietet, um auf dieser Basis dann zu diskutieren, inwieweit ein Reformbedarf bezüglich Art. 6 der RL erkennbar wird.

## B. § 69e UrhG

### I. Die Regelungen der Richtlinie und deren Umsetzung

#### 1. Das Problem

4 Tatsächlich ist das Problem, das Art. 6 der RL behandelt, eher ein kartellrechtliches. Hintergrund ist die Bedeutung von Schnittstelleninformationen für die Kompatibilität von Software und Hardware und damit die Möglichkeit, durch Geheimhaltung solcher Informationen den Wettbewerb einzuschränken. IBM war bereits in den achtziger Jahren einem Kartellverfahren ausgesetzt, bei dem es um die Offenlegung solcher Schnittstelleninformationen ging.<sup>3</sup> Vorwurf war u.a., dass IBM seine marktbeherrschende Stellung dadurch missbrauche, dass es andere Hersteller nicht rechtzeitig mit solchen Informationen versorge, die zur Herstellung wettbewerbsfähiger Produkte zu dem System von IBM notwendig waren. IBM gab dann vor dem formellen Abschluss des Verfahrens eine Verpflichtungszusage ab, die Schnittstelleninformation für das System IBM/370 und die IBM Systems Network Architecture gegenüber Wettbewerbern auf Anfrage offenzulegen, von denen diese auch rege Gebrauch machten.<sup>4</sup>

5 Betrachten wir die Problematik aus der Sichtweise des Urheberrechts, so geht es um die Problematik des sog. Reverse Engineering. „Proprietäre“ Softwarehersteller versuchen ihr Programm-Know-how dadurch zu schützen, dass sie das Programm nur in maschinenlesbarer Fassung vertreiben (Object Code). Eine Erschließung des Know-hows durch Wettbewerber ist dann nur durch eine Rückwärtsentwicklung („Reverse Engineering“) oder auch Dekompilierung möglich. Zur Terminologie: Dekompilieren bezeichnet die Rückübersetzung des Objektformats in einen Source Code; Reverse Engineering ist weiter und bezeichnet daneben auch andere Techniken der Rückwärtsanalyse.<sup>5</sup> Trotz Verfügbarkeit entsprechender Tools ist bei komplexeren Programmen immer noch ein erheblicher Aufwand nötig, um die Dekompilierung durchzuführen, wobei dieser oft größer ist als die zur Entwicklung des Originalprogramms.<sup>6</sup> Ein weiteres kommt hinzu: es ist nie möglich, den Quellcode vollständig wieder herzustellen. Vielmehr gehen jedenfalls Kommentare und ursprüngliche Formatierung verloren.

6 Weit verbreitet sind heute JAVA-Anwendungen. Wegen der Besonderheiten der Programmierumgebung war es relativ einfach, einen Großteil des Source Code mittels Dekompilierung zu rekonstruieren. Daraufhin haben die Programmierer sog. „ob-

fuscators“ (Verdunkler, Verschleierer) eingebaut, um das Dekompilieren zu erschweren.<sup>7</sup>

7 Darüber hinaus sind die Einsatzmöglichkeiten des Dekompilierens keineswegs beschränkt auf die Zwecke der Erlangung von Schnittstelleninformationen. Im Rahmen der Softwarepflege etwa kann zu Zwecken der Migration oder bei Verlust des Source Code zur Fehlerberichtigung eine Dekompilierung erforderlich sein. Auch zu Sicherheitszwecken wird die Dekompilierung eingesetzt.

8 Allen von uns ist noch das berühmte „Jahr-2000-Problem“ in Erinnerung, zu dem einige dicke Bücher geschrieben worden sind, allerdings nur im Vorfeld. Zur Bewältigung des Problems war es notwendig, die Stellen im Programm zu finden, wo ein Datum zu verarbeiten war, und festzustellen, ob zwei oder vier Stellen für die Jahreszahl vorgesehen waren.<sup>8</sup> Manche Programme waren nur noch im Objektformat verfügbar, bei manchen war die Dokumentation schlecht oder die Programmiersprache veraltet. Hier war dann das Dekompilieren ein Mittel der Abhilfe.

9 Urheberrechtlich betrachtet handelt es sich beim Dekompilieren um eine Vervielfältigung oder Bearbeitung, die eine Zustimmung des Rechteinhabers notwendig macht.<sup>9</sup>

#### 2. Die Lösung

10 im Gesetzgebungsverfahren wollte man das Problem zunächst dem Kartellrecht überlassen. Auf Intervention des Europäischen Parlaments hat dann vorgeschlagen, auch eine entsprechende Vorschrift ins Urheberrecht einzufügen,<sup>10</sup> und dies hat sich im Ergebnis auch durchsetzen können. Es gibt im Immaterialgüterrecht übrigens eine Reihe eher kartellrechtlich motivierter Ausnahmeregelungen, die flankierend zum Kartellrecht wirken sollen.

11 Zunächst hatte man an eine weite Dekompilierungs-erlaubnis gedacht, die das Dekompilieren auch zu Wartungszwecken sowie zur Erlangung kompatibler Hardware zustimmungsfrei machen sollte. Die endgültige Regelung wurde dann aber eng gefasst und auf den Zweck der Herstellung von Interoperabilität mit anderen Programmen begrenzt. Interessanterweise wurde übrigens während der Beratungen auch eine Fair-Use-Klausel als Alternative diskutiert aber wegen der abweichenden kontinentaleuropäischen Tradition zugunsten einer detaillierten Regelung wieder verworfen.<sup>11</sup>

12 Auf Seiten der Softwarehersteller zeigten sich durchaus unterschiedliche Interessenlagen.<sup>12</sup> Teilweise (von Seiten einiger großer Unternehmen der Branche) wurde auf die Gefahr der Verbreitung geheimen Know-hows und eine sich daraus ergebende Beeinträchtigung der Anreizfunktion des Urheberrechts

hingewiesen. Andere Teile der Softwareindustrie (KMU) sahen die Gefahr eher in einer Beeinträchtigung des Wettbewerbs durch fehlenden Zugang zu den notwendigen Schnittstelleninformationen. Die Anwenderseite hat teilweise noch weitergehende Zugangsinteressen zum Source Code, etwa hinsichtlich des Bedürfnisses nach Änderungen und war deshalb auch gegen die Regelung.

### 3. Die Regelung im Einzelnen

- 13** Die dann erlassene Regelung ist recht komplex und immer noch mit einigen Streitfragen zur Auslegung behaftet. Sie ist unterteilt in die Voraussetzungen der Dekompilierungsfreiheit nach Abs. 1 und die beschränkten Verwendungsmöglichkeiten der gewonnenen Informationen nach Abs. 2. Voraussetzungen für das zustimmungsfreie Dekompilieren sind nach Abs. 1:
- ▶ Unerlässlichkeit zur Herstellung von Interoperabilität, beschränkt auf die dafür notwendigen Teile
  - ▶ das zu schaffende interoperable Programm muss eine unabhängige Schöpfung darstellen
  - ▶ zulässig nur durch Lizenznehmer oder andere zur Verwendung eines Vervielfältigungsstücks berechnigte Person
  - ▶ Informationen dürfen dem Berechtigten nicht anderweitig zugänglich gemacht sein.
- 14** Der zentrale Zweck ist die Herstellung von Interoperabilität. Daher ist es wichtig, dies ein wenig zu konkretisieren. Interoperabilität bezeichnet die „funktionale Verbindung und Interaktion in logischer, funktionaler und zum Teil in physischer Hinsicht über sog. Schnittstellen“.<sup>13</sup> Dabei geht es um die „Interaktion“ zwischen Computerprogrammen ebenso wie zwischen Software und Hardware. Anerkannt ist auch, dass es nicht etwa nur um die Interoperabilität von Betriebssystemen und Anwendungsprogrammen geht, sondern das zweite Programm kann auch in Konkurrenz zum dekompierten Programm treten.<sup>14</sup> Das bedeutet andererseits nicht, dass die Wettbewerber identische Programme entwickeln dürfen; das wäre auch mit dem Zugang nur zu den Schnittstelleninformationen gar nicht möglich. Vielmehr wird es Unterschiede hinsichtlich Geschwindigkeit, Verlässlichkeit, Sicherheit und sonstiger Funktion geben. Allerdings muss etwa ein konkurrierendes Arbeitsgruppen-Betriebssystem, wie im Microsoft-Fall, in der Lage sein, bestimmte Informationen mit einem Windows Client-PC auszutauschen, also zu senden und zu empfangen.<sup>15</sup> Dazu ist es aber nicht notwendig, dass das konkurrierende Betriebssystem intern genau identisch zu dem Microsoft-Betriebssystem funktioniert.
- 15** Nach der Gesetz gewordenen Fassung sind andere Zwecke nicht zugelassen. Das betrifft auch die angrenzende Vorschrift des § 69d Abs. 1 UrhG, wonach die Fehlerberichtigung zulässig ist.<sup>16</sup> Insoweit wird § 69e als vorrangig angesehen. Anerkannt ist aber auch, dass das Merkmal der Erforderlichkeit nicht bedeutet, dass sich die Dekompilierung auf die eigentlichen Schnittstelleninformationen beschränken muss, denn deren Anordnung ist nicht immer von vornherein bekannt. Daher können auch andere Programmteile dekompiert werden, wenn und solange nicht erkennbar ist, wo sich die Schnittstelleninformationen befinden.<sup>17</sup> Zur Verhinderung von Missbrauch wird man hier aber zurückhaltend operieren müssen.
- 16** Die Verwendung der so gewonnenen Informationen ist durch Abs. 2 in dreierlei Hinsicht beschränkt:
- ▶ keine Verwendung zu anderen Zwecken
  - ▶ keine Weitergabe an Dritte
  - ▶ keine Verletzung der Rechte am Ursprungsprogramm.
- 17** Das zweite Merkmal bedeutet, dass eine Veröffentlichung der Informationen unzulässig ist. Gleiches gilt für die Weitergabe der Informationen an andere Programmhersteller, die dann selbst wieder eine Dekompilierung vornehmen müssen. Dabei geht es um den Schutz vor einer unkontrollierten Verbreitung der Informationen. Andererseits erschwert diese enge Fassung aber den Aufwand gerade für KMU.<sup>18</sup>
- 18** Probleme wirft aber vor allem der letzte Punkt auf. Zu beachten ist auch im Rahmen des § 69e die Unterscheidung von schutzfähigem Ausdruck und ungeschützter Idee. Standardisierte Spezifikationen können in verschiedenen Formen geschützten Ausdrucks also Programmcode umgesetzt werden. Insofern wird vor allem von Blocher vertreten, dass die Verwendungsbeschränkungen von vornherein nur Einschränkungen des Urheberrechts beabsichtigen, und daher auf geschützten Ausdruck beschränkt sind, während allgemeine Ideen und Prinzipien nicht erfasst werden.<sup>19</sup> Hier muss man auch Art. 1 Abs. 2 RL berücksichtigen, der ja ausdrücklich noch einmal betont, dass Ideen und Prinzipien, die den Schnittstellen zugrunde liegen, nicht geschützt sind (§ 69a Abs. 2 S. 2 UrhG). Ein solches Ergebnis erscheint aber mit der h.M. im Wege der Auslegung schwer zu erreichen.
- 19** Streitig diskutiert wird auch die Frage, ob Schnittstelleninformationen auch in einem solchen Umfang übernommen werden dürfen, dass es sich um eigentlich geschützten Ausdruck handelt, soweit dies zur Herstellung der Interoperabilität notwendig ist. Dies wird überwiegend befürwortet und dabei verwiesen auf fehlende Gestaltungsmöglichkeiten und auch die

U.S.-amerikanische „Merger“-Doktrin, die in solchen Fällen die Übernahme von Code zulässt.<sup>20</sup> Inzwischen hat sich die Programmieretechnik so weiterentwickelt, dass es bei manchen Programmen nicht mehr ohne weiteres möglich ist, gezielt die Schnittstelleninformationen isoliert zu gewinnen.<sup>21</sup> Hier ist dann verstärkt die Frage aufgeworfen, ob auch ein größerer Teil des geschützten Codes frei übernommen werden kann, um Interoperabilität herzustellen. Dies hängt natürlich vom Umfang im Einzelfall ab. Jedenfalls erscheint dies nicht mehr unbedingt von der Intention des Gesetzgebers gedeckt. Die Grenze ist jedenfalls durch § 69e Abs. 2 Nr. 3 UrhG gezogen, wenn die Übernahme zu einem Programm mit „im wesentlichen“ ähnlicher Ausdrucksform führt.

## II. Hauptkritikpunkt: Schutz von Ideen

### 1. Urheberrechtlicher Know-how-Schutz

20 Gerade der erste Punkt hat bereits während der Richtlinienentstehung zu Kritik herausgefordert.<sup>22</sup> Folgt man der weit aus h.M., so wird wegen der notwendigen Urheberrechtsverletzung zu deren Erlangung auch der Zugang zu ungeschützten Ideen beschränkt. Man hat die Vorschrift daher auch als „urheberrechtlichen Know-how-Schutz“<sup>23</sup> bezeichnet.

21 Dies scheint mir einer der kritischsten Punkte der Regelung zu sein. Das Urheberrecht versperrt den Weg zur Erlangung des Programm-Know-how. Natürlich kann man an eine sog. Black-Box-Analyse denken, die aber auch nur innerhalb der engen Grenzen von § 69d Abs. 3 UrhG zugelassen ist und meist nicht weiterführt.<sup>24</sup> Nun geht es bei dem Schutz von Know-how eigentlich um den Schutz eines Wettbewerbsvorsprungs im Interesse der Innovation. Das ist aber nicht die Stoßrichtung des Urheberrechts, dem es um die Kreativität geht, während das Partizipationsinteresse durch die reine Analysetätigkeit nicht berührt ist.<sup>25</sup> Dies hat etwa die Bundesregierung im Umsetzungsverfahren bemängelt und die Vorschrift insoweit als systemwidrig bezeichnet.<sup>26</sup>

22 Darüber hinaus ist ein so weitgehender Schutz des Know-how auch inhaltlich unangemessen. Schaut man in die technischen Schutzrechte, die ja primär für den Schutz technischen Know-how „zuständig“ sind, so bestehen etwa im Patentrecht Ausnahmen für private Nutzung sowie vor allem für Versuchszwecke, die auch für gewerbliche Zweck zugelassen sind (§ 11 Nr. 1, 2 PatG). Darüber hinaus ist es für den patentrechtlichen Schutzzweck der Innovationsförderung von essentieller Bedeutung, dass die Erfindung offenbart wird, also das technische Wissen der Allgemeinheit zur Verfügung gestellt wird, damit an-

dere Entwickler darauf aufbauen können. Instruktiv ist auch ein Blick in das Halbleiterschutzgesetz.<sup>27</sup> Nach § 6 Abs. 2 sind folgende Handlungen nicht vom Schutz umfasst:

1. die Nachbildung der Topographie zum Zwecke der Analyse, der Bewertung oder der Ausbildung;
2. die geschäftliche Verwertung einer Topographie, die das Ergebnis einer Analyse oder Bewertung nach Nummer 2 ist und Eigenart im Sinne von § 1 Abs. 2 aufweist.

23 In den technischen Schutzrechten wird das Interesse am Informationsaustausch in Wissenschaft und Forschung damit besonders berücksichtigt. Anders als der Patentschutz ist das Urheberrecht auf diesen Interessenkonflikt nicht zugeschnitten. Die Sondervorschriften zum Softwareschutz schießen noch stärker über das Ziel hinaus und sind enthalten insofern weit keinen angemessenen Interessenausgleich. Das zeigt sich auch an den Wertungen des wettbewerbsrechtlichen Know-how-Schutzes.

### 2. Wettbewerbsrechtlicher Know-how-Schutz von Schnittstellen

24 Nur im Objektformat vertriebene Programme können sich den Schutz nach §§ 17, 18 UWG zunutze machen, der sich auch gegen ein nicht genehmigtes Reverse Engineering richten kann.<sup>28</sup> Fraglich ist, ob die Wertung des Art. 6 der RL auch auf den Know-how-Schutz übergreift. Teilweise wird dies wegen der Parallelität beider Schutzmöglichkeiten und der ausdrücklichen Entscheidung des Rates gegen einen Vorrang urheberrechtlicher Bestimmungen abgelehnt wird,<sup>29</sup> Überwiegend wird aber ein Verbot der nach Art. 6 erlaubten Handlungen auf der Grundlage des Geheimnisschutzes abgelehnt mit Hinweis darauf, dass es sich bei der RL um die speziellere Vorschrift handelt.<sup>30</sup> Letzterem wird man wohl heute folgen müssen, so dass im Umfang der urheberrechtlichen Freistellung auch ein Know-how-Schutz nicht zum Tragen kommt.

25 Interessant ist insofern aber ein rechtsvergleichender Blick in die U.S.A. Dort hat das Trade Secret Law einen ungleich höheren Stellenwert als Schutzsystem erlangt als in Deutschland.<sup>31</sup> Anders als bei uns ist dort aber das reverse engineering von Computerprogrammen vom Schutz freigestellt. Tragendes Argument dafür ist, dass dieser Vorgang so zeitaufwändig und teuer ist, dass sich ein Wettbewerber dadurch keinen Wettbewerbsvorsprung verschaffen kann.<sup>32</sup> Darin liegt auch eine andere und wohl realistischere Bewertung des Interessenkonflikts zwischen Rechteinhaber und Zugangsinteressen der Wettbewerber. Die Gefährdung der ersten Gruppe, die der engen Fassung der RL zugrunde liegt, wird hier als viel geringer eingeschätzt. Dafür werden die

Interessen am Zugang zum Programm-Know-how höher bewertet. Der freie Zugang fördert Innovation und letztlich auch die Interessen der Verbraucher.<sup>33</sup>

### 3. Freigabe des Reverse Engineering

- 26 Folgt man dieser Bewertung, und angesichts der technischen und wirtschaftlichen Ausgestaltung des Reverse Engineerings muss man das wohl, so erscheint als angemessener Interessenausgleich nur eine völlige Freigabe des Reverse Engineering. Das wettbewerblich motivierte Interesse der Rechteinhaber würde dadurch nicht beeinträchtigt. Zugleich würde dem urheberrechtlichen Prinzip des freien Zugangs zu Ideen Rechnung getragen. Damit kämen die Interessen von Wissenschaft und Forschung am freien Austausch von Ideen zum Tragen.
- 27 Eine solche Freigabe öffnete den Weg auch für weitere legitime Zwecke.<sup>34</sup> Das gilt etwa für die IT-Sicherheit. Beispielsweise kann das Dekompilieren dazu dienen, zu überprüfen, ob sicherheitsrelevante Anwendungen wie Verschlüsselungssysteme frei von Trojanern oder Umgehungsmöglichkeiten sind.<sup>35</sup> Weitere sinnvolle Anwendungen ergeben sich im Bereich der Wartung und Pflege von Software. Weiterhin könnte die Dekompilierung eingesetzt werden, um den Nachweis einer Rechtsverletzung durch einen Nachprogrammierung durch einen Dritten zu ermöglichen. Dadurch könnte die Einschaltung eines unabhängigen Gutachters vermieden werden.<sup>36</sup>
- 28 Rückwirkungen ergäben sich durch eine Freigabe des Reverse Engineering auf § 69d UrhG. Dieses wäre auch zu Zwecken der Fehlerbeseitigung zulässig und könnte dieser Regelung zusätzliche Wirksamkeit verschaffen.
- 29 Zum Schutz der Interessen der Rechteinhaber sollte es aber hinsichtlich der Verwendungsbeschränkungen nach § 69e Abs. 2 UrhG dabei bleiben, dass die Schaffung von Konkurrenzprogrammen mit gleicher oder ähnlicher Ausdrucksform untersagt ist. Die ergibt sich aber bereits aus §§ 16, 23 UrhG. Andererseits besteht angesichts des Gesagten keine Notwendigkeit, den Schutz gegen die Verwendung der erlangten Information auf die Phasen der Entwicklung und Herstellung vorzuverlagern, wie es § 69e Abs. 2 Nr. 3 UrhG vorsieht.

### C. Kartellrecht als zweite Säule

- 30 Es herrscht Einigkeit, dass es sich bei § 69e um eine zumindest kartellrechtlich motivierte Vorschrift handelt. Daher ist es zur Bewertung der Regelung auch notwendig, einen Blick auf Eingreifen und Reichweite des eigentlichen Kartell- und Wettbe-

werbsrechts zu werfen. Das kann zur Klärung beitragen, inwieweit überhaupt ein Bedürfnis für eine spezielle Ausnahmevorschrift zur Herstellung von Interoperabilität im Urheberrecht besteht. Hier geht es vor allem um das Missbrauchsverbot des Art. 102 EG (vormals 82). Dabei können wir das Missbrauchsverfahren gegen Microsoft heranziehen, das mit der Entscheidung des EuG vom 17.9.2007 endete<sup>37</sup> und in dem zuvor die Kommission ein Bußgeld in bis dato unerreichter Höhe von 497 Mio € festgesetzt hatte.

### I. Das Verfahren Kommission gg. Microsoft 2004/2007

- 31 Die Kommission hatte sich auf Art. 82 lit b EG a.F. gestützt, wonach ein Missbrauch insbesondere in der Einschränkung der Erzeugung, des Absatzes oder der technischen Entwicklung zum Schaden der Verbraucher bestehen kann. Im Kern ging es in dem für uns relevanten Teil des Verfahrens um den Vorwurf, dass Microsoft konkurrierenden Herstellern von Arbeitsgruppen-Serverbetriebssystemen nicht alle Schnittstelleninformationen zur Verfügung stellte, die diese zur Herstellung „voller Interoperabilität“ mit Windows-basierten Client-PC's benötigten.<sup>38</sup> Ziel sei es, die überragende Stellung auf dem Markt für PC-Betriebssysteme auszunutzen, um Wettbewerber von den Märkten für Serverprogramme zu verdrängen. Microsoft hatte zwar immer in gewissem Umfang Schnittstelleninformationen den Wettbewerbern zur Verfügung gestellt, nicht aber für bestimmte neue Windows-Funktionen, z.B. Active Directory. Insoweit geschlossene Vergleiche in den U.S.A. reichten der Kommission nicht aus.
- 32 Kernfrage war insoweit, wann das Kartellrecht der Ausübung immaterialgüterrechtlicher Befugnisse Grenzen setzen kann. Der EuGH hatte bereits in einigen vorhergehenden Entscheidungen Pflöcke für die ausnahmsweise Annahme eines Missbrauchs von Immaterialgüterrechten eingeschlagen, vor allem in den Entscheidungen Magill<sup>39</sup> und IMS Health<sup>40</sup>. Voraussetzung ist danach, dass
1. die Lizenzierung unerlässlich ist für den Zugang zu einem benachbarten Markt
  2. ihre Verweigerung jeden wirksamen Wettbewerb ausschließen würde
  3. das Erscheinen eines neuen Produkts auf diesem Markt verhindern würde
  4. keine objektive Rechtfertigung vorliegt.
- 33 Die zu diesen Voraussetzungen in diesem Verfahren ausgetauschten Argumente erinnern an die Punkte, die auch in der Diskussion zu § 69e UrhG von besonderer Bedeutung sind. Microsoft trug vor, es fehle

an der Unerlässlichkeit der Preisgabe der Informationen, da diese bereits auf fünf anderen Wegen von Wettbewerbern beschafft werden könnten. Das EuG hob jedoch hervor, dass Interoperabilität für die Endkunden eine große Bedeutung habe und daher eine den MS-Produkten vergleichbare Interoperabilität hergestellt werden müsse, um wettbewerbsfähig zu sein.

- 34 Diese strengen Anforderungen sind kritisiert worden. Hier wird darauf verwiesen, der EuGH habe bisher keine völlige Gleichstellung verlangt, sondern bereits das Bestehen weniger günstiger Alternativen ausreichen lassen.<sup>41</sup> Andererseits muss man hier wohl auch die Besonderheiten bei der Software berücksichtigen, die für die Notwendigkeit einer „vollen“ Interoperabilität sprechen.
- 35 Hinsichtlich der zweiten Voraussetzung trug Microsoft vor, es würde keinesfalls jeglicher Wettbewerb durch die Verweigerung ausgeschlossen. Das Gericht betonte hierzu, das die Märkte besonders durch Netzwerkeffekte geprägt seien und ein Ausschluss vom Wettbewerb nicht rückgängig zu machen sei. Auch sei der Marktanteil von Microsoft stetig gewachsen.
- 36 Zum Erfordernis des „neuen Produkts“ hatte Microsoft geltend gemacht, die Wettbewerber würden die Produkte einfach klonen, so dass keine neuen Produkte entstünden. Das EuG hielt es jedoch für ausreichend, wenn die Wettbewerbsprodukte „substantielle“ eigene Elemente enthielten. Darüber hinaus wurde das Erfordernis selbst vom EuG unter Hinweis auf die allgemeinere gesetzliche Formulierung der Einschränkung der technischen Entwicklung zum Nachteil der Verbraucher angezweifelt. Letzteres sei hier gegeben, da die Wettbewerber durch die Bindung der Kunden an Microsoft-Produkte ihre innovativen Produkte nicht absetzen könnten.
- 37 Im Rahmen einer Rechtfertigung prüfte das Gericht dann das Argument, die Zwangslizenzierung führe zur Herabsetzung der Innovationsbereitschaft bei Microsoft und anderen Wettbewerbern. Dies sah das Gericht aber nicht als hinreichend dargelegt an.
- 38 Während in der Entscheidung des EuG Art. 6 der SoftwareRL nur mittelbar von Bedeutung war, hatte der Kommissionsbeschluss die Richtlinie ausdrücklich angesprochen.<sup>42</sup> Microsoft hatte dazu vorgetragen, dass die Entscheidung den ausgewogenen Ausgleich in Frage stelle, den die RL zwischen Urheberrecht und Wettbewerbspolitik hergestellt habe. Die Kommission stellte dazu fest, dass die RL als sekundäres Gemeinschaftsrecht hinter Art. 82 EG zurücktreten müsse. Darüber hinaus sei Art. 6 ein guter Ausgangspunkt auch für die kartellrechtliche Bewertung und stimme inhaltlich mit den von der Kommission gestellten Anforderungen überein. Allerdings gehe Art. 82 insofern über die SoftwareRL hinaus, als

hier nicht nur eine passive Duldungspflicht hinsichtlich der Dekompilierung begründet werde, sondern eine aktive Pflicht zur Bereitstellung der entsprechenden Informationen. Erwägungsgrund 27 der RL lasse aber ausdrücklich die Anwendung der Wettbewerbsregeln des EGV unberührt.

- 39 Eingehend diskutiert wurden auch Microsofts Argumente, die gestellten Anforderungen gingen über das hinaus, was Art. 6 RL im Hinblick auf Interoperabilität fordere. Demgegenüber bekräftigten die Kommission und danach auch das EuG, dass die in dem Verfahren gestellten Anforderungen mit dem Erfordernis der Interoperabilität in der RL kongruent seien. Vor allem beinhalte dieses das Vorhandensein der Möglichkeit zum wechselseitigen Austausch und Nutzung der Informationen. Microsofts engere Definition würde demgegenüber bedeuten, dass bereits jeder noch so beschränkte Zugang eines Clients zum Server-Betriebssystem ausreichen würde, um hinreichend Interoperabilität herzustellen.

## II. Bewertung

- 40 Will man insoweit die komplementär gegebenen kartellrechtlichen Möglichkeiten bewerten, so zeigt sich, dass diese inhaltlich weitgehend mit den Anforderungen des Art. 6 konsistent sind, auch wenn zusätzlich wettbewerbsrechtliche Kriterien zu prüfen sind. Beschränkt wird der Anwendungsbereich des Art. 82 EG durch das Erfordernis der marktbeherrschenden Stellung, während das Urheberrecht auf alle Hersteller Anwendung findet. Hier lässt sich allerdings fragen, ob der auch Art. 6 zugrundeliegende Zweck es nicht auch rechtfertigt, die Vorschrift nur auf marktbeherrschende Unternehmen anzuwenden, da nur von diesen eine Gefahr für den Wettbewerb ausgeht und kleinere Hersteller eher geschützt werden müssten.
- 41 Weiterhin ist es zwar grundsätzlich richtig, dass Art. 6 nur eine passive Duldungspflicht begründet, er zielt aber auch darauf ab, die Unternehmen zu einer frühzeitigen Veröffentlichung zu veranlassen, um die bei der Dekompilierung immer gegebene Missbrauchsmöglichkeit von vornherein auszuschließen. Insofern bestehen in der Zielrichtung keine wesentlichen Unterschiede.
- 42 Zu beachten bleibt aber, dass das Reverse Engineering einen zeit- und kostenaufwändigen Prozess beinhaltet. Schließlich stehen dem Hersteller auch andere Möglichkeiten zur Verfügung, um die Herstellung von Kompatibilität zu erschweren. Dazu gehört ein Upgrade des Betriebssystems, wie etwa beim Übergang von Windows NT auf Windows 2000. Das war auch der Grund, warum die Fa. Sun sich im Verfahren vor der Kommission nicht auf das reverse engineering verlassen hat sondern aktive Offenlegung der Schnittstellen gefordert hat. Dies scheint also das

wesentlich effektivere Mittel zu sein, um den Wettbewerb zu schützen. Es vermeidet auch unnötigen Parallelaufwand.

- 43 Hier scheint mir einer der wesentlichen Gründe zu liegen, warum Art. 6 RL scheinbar in der Praxis bisher kaum eine Rolle spielt. Andererseits wäre eine aktive Offenlegungspflicht für das Urheberrecht ein Fremdkörper, im Kartellrecht dagegen am richtigen Platz.<sup>43</sup>

## D. IV. Schlussfolgerungen

- 44 Der letztere Aspekt ist von großer Bedeutung auch für ein Gesamtfazit unter dem Gesichtspunkt eines Reformbedarfs. Natürlich ist – wie nicht selten – die sprachliche Fassung der Richtlinie nicht immer glücklich und an manchen Stellen verbesserungs- und klärungsbedürftig. Auch habe ich einige Streitfragen zur Reichweite der Erlaubnis angesprochen, die man von Seiten des Gesetzgebers klarstellen könnte. Dazu gehören die Zulässigkeit der Dekompilierung zur Herstellung der Interoperabilität von Hardware sowie die Erlaubnis zur Übernahme eigentlich urheberrechtlich geschützter Code-Elemente, soweit dies zur Herstellung der Interoperabilität unvermeidlich ist.

- 45 Ich möchte aber den Schwerpunkt auf eine weiter gefasste Perspektive legen. Die Ausgangsfrage lautet: „Reichen die Vorschriften der Richtlinie“. Die Frage ist natürlich: reichen wozu? Wenn es um den Zweck eines angemessenen Ausgleichs zwischen Urheberrechtsschutz und Wettbewerb geht, so komme ich zu dem Ergebnis, dass die Ansiedlung dieser Vorschrift im Urheberrecht ein Fehlgriff war. Dafür sprechen nicht nur die bisher nicht erkennbare praktische Relevanz, sondern auch der Vergleich zum komplementären Kartellrecht. Daraus möchte ich folgende Thesen/Forderungen ableiten:

1. *Reverse Engineering urheberrechtlich ganz freigeben!*

- 46 Hierfür sind mehrere Erwägungen ausschlaggebend:

- 47 Die von Blocher zu Recht in den Mittelpunkt gestellte Kritik am systemfremden Schutz der Ideen durch das Urheberrecht, die sich allerdings im Wege der Auslegung nicht beseitigen lässt. Hier könnte eine Befreiung der Dekompilierungserlaubnis von den engen Grenzen der Interoperabilität helfen, dem Prinzip der Schutzfreiheit der Ideen wieder zur Geltung zu verhelfen;

- 48 Die Freigabe der Zwecke ermöglicht die Dekompilierung auch zu Zwecken von Forschung und Entwicklung sowie weitere Zwecke, etwa der IT-Sicherheit. Damit würde man den Gleichklang zu den Wertungen der technischen Schutzrechte herstellen, von denen man keineswegs sagen kann, dass sie die Her-

steller schädigen; vielmehr wird der Informationsaustausch durch das Einzwängen des Softwareschutzes in das Urheberrecht noch stärker behindert, als wenn man Software in den technischen Schutzrechten belassen hätte; zugleich bekäme Art. 6 damit stärker den Charakter einer echten Schrankenregelung;

- 49 eine unangemessene Gefährdung der Interessen der Rechteinhaber wäre damit nicht verbunden: bei den Programmen, um die es hier geht, ist der Prozess des reverse engineering so zeitaufwändig, dass ein Verlust der für die Innovation notwendigen lead time für die Hersteller nicht zu befürchten ist; im übrigen ist ein Schutz dieser Lead time nicht Aufgabe des Urheberrechts, sondern gehört in das Wettbewerbsrecht; nicht ohne Grund ist aber auch im amerikanischen relativ strengen Trade Secret Law das reverse engineering auch bei Software erlaubt, da sich der Dekompilierer dadurch keinen Wettbewerbsvorsprung verschafft; das leitet über zum nächsten Punkt

2. *Wettbewerbsaspekte sind besser im Wettbewerbsrecht aufgehoben!*

- 50 Das Kartellrecht ist von Eingriffsschwelle und inhaltlicher Ausgestaltung sehr viel besser geeignet, die Gefährdung des Wettbewerbs zu erfassen und angemessen zu bekämpfen. Die Kommission hat mittlerweile auch eindrucksvoll ihren Willen gezeigt, dies auch in die Praxis umzusetzen. Auch die Rechtsfolge einer aktiven Informationspflicht wird dem Wettbewerbsgedanken sehr viel effektiver gerecht als die bloß passive Duldungspflicht hinsichtlich des Kompilierens.

- 51 Diese Überlegungen sind nicht neu und haben zT schon in der Diskussion im Vorfeld der Richtlinie eine große Rolle gespielt. Die Tatsache, dass die Vorschrift weitgehend totes Recht ist, während andererseits das Kartellrecht immer größere Bedeutung erlangt, sollte uns aus heutiger Sicht ein neuer Anstoß sein, die Regelung zu überdenken. Nicht zuletzt auch die zunehmende Strömung in Richtung auf mehr Access im Urheberrecht im Interesse von Wissenschaft, Forschung und Informationsfreiheit gibt uns heute zusätzliche Argumente, die Vorschrift jedenfalls zu lockern.

1 Vgl. Dreier/Schulze-Dreier § 69e Rn. 7.

2 Vgl. a.a.O.

3 EuGH, Rs. 60/81, 190/81 – IBM/Kommission, Slg. 1981, 2639.

4 Vgl. Loewenheim/Meessen/Riesenkampff, Kartellrecht, 2. Aufl. München 2009, GRUR Rn. 126; Sucker, CR 1988, 271, 274 ff.

5 Vgl. Harte-Bavendamm, GRUR 1990, 657 ff. Zur Zulässigkeit Schrickler/Loewenheim, § 69d Rn. 22 f.

6 Eingehend dazu Blocher/Walter, in: Walter/v. Lewinsky (eds.), European Copyright Law, 2nd ed., Oxford/New York 2010, Computer Program Directive Zi. 5.6.13.

- 7 Vgl. a.a.O., Zi. 5.6.20.
- 8 Vgl. Blocher/Walter (Fn. 6), Rn. 5.6.16.
- 9 Haberstumpf, CR 1991, 129, 132 ff.
- 10 ABL.EG 1990 Nr. C 231, S. 81.
- 11 Vgl. Blocher/Walter (Fn. 6) Rn. 5.6.6.
- 12 Vgl. Blocher/Walter (Fn. 6) Rn. 5.6.3; Marly, Praxishandbuch Softwarerecht, 5. Aufl. München 2009, Rn. 230 ff., jew. m.w.Nachw.; Haberstumpf, CR 1991, 129 f.
- 13 Dreier (Fn. 1) § 69e Rn. 11. Vgl. auch Erwägungsgrund 12 der RL. Zur Verwendung des Begriffs im europäischen Recht vgl. Blocher/Walter (Fn. 6) Rn. 5.6.29.
- 14 OLG Düsseldorf, CR 2001, 371, 372; Dreier (Fn. 1) Rz 11 m.w.Nachw.
- 15 Vgl. EuG 17.9.2007, Rs. T-201/04, Rn. 231 ff. – Microsoft.
- 16 Schricker/Loewenheim, 4. Aufl. München 2010, § 69e Rn. 10; Dreier (Fn. 1) § 69e Rn. 12. Konrad/Timm-Goltzsch/Ullrich, in: Ullrich/Lejeune, der Internationale Softwarevertrag, 2. Aufl., Frankfurt 2006, Rn. 823, verweisen auf die Möglichkeit der Hinterlegung von Source Code und Entwicklerdokumentation.
- 17 Vgl. auch Czarnota/Hart, Legal Protection of Computer Programs in Europe, 1991, S. 80.
- 18 Schricker/Loewenheim (Fn. 16), § 69e Rn. 20.
- 19 Blocher/Walter (Fn. 6), Rn. 5.6.41; Marly (Fn. 12) Rn. 249; a.A. wohl die h.M. vgl. Wandtke/Bullinger/Grützmaker, § 69e Rn. 20 m.w.Nachw.
- 20 Eher zustimmend Dreier (Fn. 1 Rn. 20); Vinje, GRUR Int. 2991, 250, 259 f.; Marly (Fn. 1) Rn. 250, auch unter Berufung auf die „Merger“-Doktrin; ablehnend. Wandtke/Bullinger/Grützmaker, Praxiskommentar zum Urheberrecht, 4. Aufl. München 2011, § 69e Rn. 11.
- 21 Diesen Hinweis verdanke ich Malte Grützmaker auf der Tagung „20 Jahre Computerprogramm-Richtlinie“ am 12.5. in Hannover.
- 22 Vgl. die Nachweise bei Pres, Gestaltungsformen urheberrechtlicher Softwarelizenzverträge, Köln 1994, S. 137 f.; Haberstumpf, CR 1991, 129, 140; Lehmann, CR 1989, 1057, 1060.
- 23 Marly (Fn. 12), Rn. 236; vgl. ferner Schulte, CR 1992, 648, 653; Wandtke/Bullinger/Grützmaker (Fn. 20), Rn. 3.
- 24 Danach ist nur die Anzeige des Codes auf dem Bildschirm erlaubt, nicht jedoch die Dekompilierung, vgl. Dreier (Fn. 1), § 69e Rn. 12.
- 25 Klar in dieser Hinsicht auch Haberstumpf, CR 1991, 129, 140 f.; Kindermann, CR 1990, 640.
- 26 BT-Drs. 12/4022 v. 18.12.1992, S. 13.
- 27 Vgl. auch Blocher/Walter (Fn. 1), Rn. 5.6.22.
- 28 Vgl. Wiebe, Know-How-Schutz von Computersoftware, München 1993, S. 182 ff, 267 ff.
- 29 Wiebe (Fn. 28), S. 269; Moritz, CR 1993, 257, 267.
- 30 LG Mannheim, NJW 1995, 3322, 3323; Dreier (Fn. 1), Rn. 5; Wandtke/Bullinger/Grützmaker (Fn. 1), Rn. 31; Pres (Fn. 22), S. 142.
- 31 Vgl. Wiebe (Fn. 20), S. 330 ff.
- 32 *Bonito Boats, Inc. v. Thundercraft Boats, Inc.*, 489 U.S. 141 (1989); *Kewanee Oil Co. v. Bicron Corp.*, 416 U.S. 470 (1974); *E.I. duPont de Nemours & Co., Inc. v. Christopher*, 431 F.2d 1012 (5<sup>th</sup>. Cir. 1970). Vgl. auch Wiebe (Fn. 20), S. 384 f.
- 33 Eingehend Samuelson/Scotchmer, *The Law and Economics of Reverse Engineering*, 111 Yale L. J. 1575 (2002).
- 34 Dazu auch Haberstumpf, CR 1991, 129, 130.
- 35 Vgl. Blocher/Walter (Fn. 6), Rn. 5.6.15.
- 36 Vgl. Dreier (Fn. 1), § 69e Rn. 12, der für anhängige Gerichtsverfahren auch eine Analogie zu § 45 UrhG befürwortet.
- 37 EuG 17.9.2007, RS T-201/04 – Microsoft; dazu Körber WuW 2007, 1209 ff.
- 38 Kommission, 24.3.2004, COMP/C-3/37.792 – Microsoft.
- 39 EuGH 6.4.1995, Rs. 241, 242/91, Slg. 1995, I-743 – Magill.
- 40 EuGH 29.4.2004, Rs. C-418/01, Slg. 2004, I-5039 – IMS Health.
- 41 Körber, WuW 2007, 1209, 1213, unter Verweis auf EuGH, Rs. C-7/97, Slg. 1998, I-7791 – Bronner.
- 42 S. Fn. 38 unter 5.3.1.4.1.4.
- 43 Für eine kodifizierte Zwangslizenzregelung im Kontext des Datenbankschutzes Grützmaker, *Urheber-, Leistungs- und Sui-generis-Schutz von Datenbanken*, Baden-Baden 1999, S. 380 ff.