

Softwareprobleme im Patentrecht?

von Axel Pfeiffer¹

Der Text wurde 2004 in der Juni-Ausgabe von „IFE“ (Informatik – Forschung und
5 Entwicklung), Springer-Verlag, veröffentlicht.

Inhalt:

1. Historie, Status, weiteres Verfahren

2. Rechtslage, Amtspraxis, Anmelderverhalten

10 **3. „Argumente“, Paradigmen, Sichtweisen**

4. Taktik

5. Interessenlagen

6. To do

15 Seit Jahren ist die Frage der Patentierung softwareimplementierbarer Erfindungen
in der Diskussion, und das auf allen Ebenen, nämlich in der Exekutive, der Judika-
tive und der Legislative. Den jüngsten Beitrag zu dieser Diskussion lieferte Ende
September 2003 das Europäische Parlament in Form einer Stellungnahme zum Vor-
schlag der EU-Kommission für eine Richtlinie zur Patentierbarkeit computerimp-
20 lementierter Erfindungen. Das Parlament hat dazu in nach Auffassung des Autors
bedenklicher bis absurd-zynischer Weise Stellung genommen². Es schaut also nicht
so aus, als würde die Frage der Patentierbarkeit solcher Erfindungen in absehbarer
Zeit in ruhige Gewässer gelangen. Einige Beobachtungen zu den Akteuren, Interes-
senlagen, Sichtweisen und Vorgängen der letzten Jahre seien unten zusammenge-
25 stellt. Patentrechtliche Abwägungen sind in diesem Text kaum zu finden. Insbeson-
dere des Autors detaillierte Auffassung zu rechtlichen Fragen wird hier nicht wie-

¹ Dipl.-Ing., Patentanwalt, Beetz & Partner, München, apfeiffer@beetz.com

² <http://www3.europarl.eu.int/omk/omnsapir.so/pv2?LISTING=AfficheTout&PRG=CALDOC&FILE=20030924&LANGUE=DE&TPV=PROV#Title3>

dergegeben, da es eine Wiederholung dessen wäre, was schon anderenorts³ dargestellt wurde.

5 Kurz zum Sprachgebrauch: Der Begriff „Software“ wird in diesem Text synonym zum im Patentrecht verwendeten Begriff „Programm für Datenverarbeitungsanlagen ... als solche“ benützt. Des Autors Verständnis davon ist „Zum Verständnis für eine Maschine verfasste Folge von Anweisungen“.

1. Historie, Status, weiteres Verfahren

10

1.1 Vor 1973

Bis 1972 einschließlich war Software im gewerblichen Rechtsschutz ein rechtliches Nullum in dem Sinne, dass sich das geschriebene Gesetz mit ihr nicht beschäftigte. 15 Weder das Urheberrecht noch das Patentrecht nannten Software ausdrücklich. Das war zum damaligen Zeitpunkt auch kein Wunder, da damals das Thema Software in deutlich anderen Kontexten stand als heute. Zur zeitlichen Einordnung: Ende 1972 besaß der Autor dieses Textes als 10-jähriger Schüler und frischgebackener Gymnasiast als „letzten Schrei“ einen 10*5*2 cm messenden Taschenrechner mit Leuchtdiodenanzeige, dessen höchstentwickelte Funktion die Prozenttaste war. 20 Damals sah man selbstverständlich die Hardware als das ewige Nadelöhr der Digitaltechnik an, und Software wurde als etwas begriffen, das – weil „soft“ statt „hard“ – gegebenenfalls immer leicht und schnell beigebracht werden konnte, wenn es Hardware gab, auf der sie hätte laufen können.

25

1.2 1973: Europäisches Patentübereinkommen (EPÜ)

³ GRUR 2003, Seite 581, verfügbar auch bei www.beetz.com, dort bei A. Pfeiffer

Zum Verständnis sei, weil das Thema verwirrend sein mag, zunächst ein kurzer Exkurs zu europäischen Institutionen, insbesondere Patente betreffend, gestattet:

Die *Europäische Patentorganisation* (EPO) ist eine zwischenstaatliche Einrichtung
 5 in Europa, die seit ihrer Gründung 1973 bis heute unabhängig von der Europäischen Union (EU, damals EWG, EG) existiert. Die EPO ist durch das Europäische Patentübereinkommen (EPÜ) geschaffen worden und ist Träger des Europäischen Patentamts (EPA, engl. European Patent Office, auch EPO abgekürzt). Diese Strukturen (EPO, EPÜ, EPA) stehen systematisch betrachtet *neben* der EU, nicht *unter* ihr.

10

Die *Europäische Union* (EU) versucht ihrerseits, im Großen einheitliche Patentierungsregeln zu verfassen, die weitergehend sein sollen als diejenigen des EPÜ. Das EU-Projekt läuft unter der Bezeichnung „Gemeinschaftspatent“, und diese Regelungen sollen nicht nur wie das EPÜ die *Erteilung* von Patenten regeln, sondern
 15 auch gemeinsame Regelungen zu *Patentverletzungsfragen und –verfahren* schaffen, zu denen sich das EPÜ ausschweigt. Im Elektronikslang könnte man das Gemeinschaftspatentprojekt der EU salopp als „Set top box“ zum EPÜ bezeichnen, denn in den bisher erdachten Strukturen der Gemeinschaftspatentregelungen ist vorgesehen, die konkreten Arbeiten zu Patentanmeldungen vom Europäischen Patentamt der
 20 EPO erledigen zu lassen, denn es wäre ja widersinnig, wenn sich Europa zwei nebeneinander her arbeitende Patentämter leisten würde. Wegen schwieriger prozessualer Fragen und auch wegen Sprachproblematiken ist das Gemeinschaftspatent bisher aber nur ein großes Projekt. Im Kleinen bemüht sich die EU parallel hierzu durch den Erlass von Verordnungen und Richtlinien um Rechtsangleichung in den
 25 Mitgliedsstaaten. In diesem Kontext begannen die Arbeiten an der eingangs erwähnten EU-Richtlinie zu computerimplementierten Erfindungen.

Das heutige Verhältnis zwischen EPO und EU kann man als „teils sympathiewillig“ bezeichnen. Die Organisationen betreiben aber durchaus Projekte nebeneinander

her, und völlige Konvergenz scheint auch langfristig nicht immer und überall erwünscht zu sein. Die Mitgliedsländer von EPO und EU sind nicht deckungsgleich, wenngleich man sagen kann, dass in beiden Organisationen alle großen europäischen Länder und viele kleinere auch Mitglieder sind.

5

Das Europäische Patentübereinkommen (EPÜ) wurde Anfang der 70er Jahre ausgearbeitet und im Jahre 1973 verabschiedet. Mit dem EPÜ wurde die oben genannte EPO geschaffen und das EPA etabliert. Schon damals erhielt der Art. 52 EPÜ seinen heutigen Wortlaut⁴, insbesondere auch Art 52(2)(c) und Art. 52(3), die zu den heutigen Softwarediskussionen Anlass geben. Aus heutiger Sicht (2004) ist diese
10 Regelung also inzwischen über 30 Jahre alt. Vier Jahre später – 1977 – erlebte PET 2001 seine Markteinführung.

1.3 1978: Betriebsaufnahme, Rechtsangleichung

15

Im Jahr 1978 nahm das Europäische Patentamt (EPA) seinen Betrieb auf und erteilt seitdem Patente entsprechend den Kriterien des EPÜ. Im gleichen Jahr erfolgte in Deutschland eine Anpassung materiellrechtlicher Regelungen des nationalen Patentgesetzes (PatG) an die entsprechenden des europäischen Patentübereinkommens, insbesondere dahingehend, dass die die Patentfähigkeit betreffenden Paragrafen im PatG praktisch wortgleich zu den entsprechenden Artikeln des EPÜ wurden.
20

⁴ **EPÜ Artikel 52:** Patentfähige Erfindungen

(1) Europäische Patente werden für Erfindungen erteilt, die neu sind, auf einer erfinderischen Tätigkeit beruhen und gewerblich anwendbar sind.

(2) Als Erfindungen im Sinn des Absatzes 1 werden insbesondere nicht angesehen: a) Entdeckungen sowie wissenschaftliche Theorien und mathematische Methoden; b) ästhetische Formschöpfungen; c) Pläne, Regeln und Verfahren für gedankliche Tätigkeiten, für Spiele oder für geschäftliche Tätigkeiten sowie Programme für Datenverarbeitungsanlagen; d) die Wiedergabe von Informationen.

(3) Absatz 2 steht der Patentfähigkeit der in dieser Vorschrift genannten Gegenstände oder Tätigkeiten nur insoweit entgegen, als sich die europäische Patentanmeldung oder das europäische Patent auf die genannten Gegenstände oder Tätigkeiten als solche bezieht.

(4) Verfahren zur chirurgischen oder therapeutischen Behandlung des ...

1.4 Seitdem

Trotz einer durchaus nicht einfachen und auch bis in die jüngste Zeit hinein nicht geradlinigen Genese kann man sagen, dass die Erteilungspraxis des Europäischen Patentamts und des Deutschen Patent- und Markenamts zu softwareimplementierbaren Erfindungen als über die Jahrzehnte hinweg abnehmend restriktiv bezeichnet werden kann. Dieser Weg wird durch diverse Entscheidungen der zuständigen Stellen (Beschwerdekammern des Europäischen Patentamts, Bundespatentgericht, BGH) begleitet und ist an ihnen ablesbar. Die heutige Handhabung des Softwareausschlusses von Art. 52 EPÜ ist weiter unten bei 2.1 und 2.2 näher dargelegt.

1.5 1993: Urheberrecht

Erst 1993 wurde für Deutschland das Urheberrecht in seiner heutigen Form für den Schutz von Software eindeutig abgefasst. Kurz gesagt verbietet das Urheberrecht das unbefugte Kopieren von Software, nicht aber die zufällige identische Neuprogrammierung oder die Andersprogrammierung von bestimmten Abläufen, die vorher schon in bestimmter Weise programmiert wurden. Urheberrecht liefert keinen Konzeptschutz. § 69g des Urheberrechts sagt, dass die Bestimmungen des Urheberrechts zu Software die Anwendung des Patentrechts Software betreffend unberührt lässt.

1.6 1995: TRIPS

Im Jahr 1995 wurde im Rahmen der WTO das Abkommen TRIPS („Agreement on Trade Related Aspects of Intellectual Property Rights“) verabschiedet. Art. 27⁵, dort

⁵ **TRIPS: Article 27** Patentable Subject Matter

1. Subject to the provisions of paragraphs 2 and 3, patents shall be available for any inventions, whether products or processes, in all fields of technology, provided that they are new, involve an inventive step and are capable of industrial application. Subject to paragraph 4 of Article 65, paragraph 8 of Article 70 and

insbes. Abs. 2, verpflichtet die Vertragsstaaten dazu, dafür zu sorgen, dass der Patentschutz für alle Erfindungen erhältlich ist, die technisch sind und auf erfinderischer Tätigkeit beruhen. Abs. 3 läßt Ausnahmen die Patentfähigkeit betreffend zu. Software ist hier aber nicht genannt.

5

1.7 Open-Source, Linux

Seit vielleicht Mitte der 90er Jahre etabliert sich im Bereich der Schaffung von Software ein Arbeits- und Geschäftsmodell, das unter dem Stichwort „Open-Source“ läuft. Bekanntestes Open-Source-Produkt dürfte das Betriebssystem Linux sein, das als Konkurrenzangebot zum ubiquitären Betriebssystem Windows hochwillkommen ist und weit verbreitet Sympathie genießt – auch beim Autor dieses Textes.

Die Programmerstellung bei Open-Source geschieht in der Weise, dass der Source-Code zu Programmen und Programm-Clustern öffentlich verfügbar ist, sodass alle, die willens und dazu in der Lage sind, den Source-Code ergänzen, fortschreiben, verändern und verbessern können. Um statt unkontrollierten Wildwuchses ausgehend von einer einmal geschaffenen Einheit doch eine zielgerichtete Entwicklung zu ermöglichen, gibt es faktisch anerkannte Autorisierungsstrukturen, die die verschiedenen Vorschläge der Ergänzung bzw. Änderung des Source-Codes prüfen, erwägen, testen und auswählen und von Zeit zu Zeit neue bzw. weiterentwickelte

paragraph 3 of this Article, patents shall be available and patent rights enjoyable without discrimination as to the place of invention, the field of technology and whether products are imported or locally produced.

2. Members may exclude from patentability inventions, the prevention within their territory of the commercial exploitation of which is necessary to protect ordre public or morality, including to protect human, animal or plant life or health or to avoid serious prejudice to the environment, provided that such exclusion is not made merely because the exploitation is prohibited by their law.

3. Members may also exclude from patentability:

(a) diagnostic, therapeutic and surgical methods for the treatment of humans or animals;

(b) plants and animals other than micro-organisms, and essentially biological processes for the production of plants or animals other than non-biological and microbiological processes. However, Members shall provide for the protection of plant varieties either by patents or by an effective sui generis system or by any combination thereof. The provisions of this subparagraph shall be reviewed four years after the date of entry into force of the WTO Agreement.

Versionen eines Programms als aktuelle Version deklarieren, festlegen und dann auch zum Download vorhalten. Ausgehend von dieser durch Deklaration ausgezeichneten und verfügbar gemachten Version können dann Fortentwicklungen vorgenommen werden. Der so weitergeschriebene Source-Code steht regelmäßig unentgeltlich zur Verfügung. Privat- und Firmenanwender können ihn übernehmen, verwenden und für eigene Zwecke modifizieren. Das geschäftliche Interesse der Firmen findet zumindest auch im Customizing des Source-Codes und im Serviceangebot um den Source-Code herum seine Befriedigung.

10 1.8 Ab Ende der 90er: Widerstand

Ab 1998 lässt sich organisierter Widerstand gegen die Patentierung softwareimplementierbarer Erfindungen feststellen. Sicher nicht ausschließlich, aber zumindest auch - und nach Auffassung des Autors überwiegend - kommt dieser Widerstand aus Open-Source-Kreisen. Anstelle vieler Worte sei per Fußnote⁶ schlicht auf einige Internetadressen verwiesen. Zumindest Teile dieser Strukturen und Organisationen dürften heute nicht mehr nur als Interessenvereinigungen anzusehen sein, sondern auch als hocheffiziente und erfahrene Lobbyarbeiter, besonders in Europa und ganz besonders in Deutschland.

20

1.9 Ab ca. 2000: Bemühungen der Europäischen Patentorganisation (EPO)

Im Jahr 2000 nahm angesichts der immer wieder auftretenden Frage nach der Patentierbarkeit softwareimplementierbarer Erfindungen die EPO einen Anlauf dahingehend, den 1973 verabschiedeten Artikel 52 EPÜ in der Weise zu modernisieren, dass zur Vermeidung von Missverständnissen der faktisch ins Leere laufende Softwareausschluss – siehe weiter unten bei 2.1 – gestrichen wird. Beim formalen Zusammentritt der die EPO bildenden Regierungskonferenz in München im Frühjahr

2001 wurde der entsprechende Vorschlag jedoch abgelehnt. Die Revision war damit insoweit gescheitert, und es blieb bei den bestehenden Regelungen.

1.10 Ab ca. 2000: Bemühungen der EU

5

Nachdem die von der EU unabhängige EPO in ihrem Bemühen, die Softwareregulungen zu aktualisieren, gescheitert war, nahm sich die EU-Kommission verstärkt der Sache an, sie war damit allerdings auch schon vorher befasst. Im Frühjahr 2002 kam es zur Vorlage eines Entwurfs einer EU-Richtlinie, der zwiespältig aufgenommen wurde. Seitens der meisten professionell im Patentwesen Tätigen wurde der Entwurf im Wesentlichen wohlwollend zur Kenntnis genommen. Die Gegner von Patenten auf softwareimpelmentierbare Erfindungen liefen allerdings Sturm dagegen. Schon die Abfassung des Richtlinienvorschlags durch die Kommission versuchte man zu beeinflussen, allerdings mit geringem Erfolg. Nachdem der Vorschlag Anfang 2002 offen gelegt wurde und es absehbar wurde, dass das Europäische Parlament damit befasst wird, setzte die Bearbeitung der jeweils nationalen Vertreter ein. Es gab E-Mail-Aktionen, Demonstrationen, Diskussionsrunden, Pressegespräche, Petitionen, Web-Streiks und Ähnliches, die dieser Interessengruppe zu staunenswertem Presseecho und Gehör in den Reihen der europäischen Parlamentarier verhalfen. Statements von drohender Patentierung von Jazz-Rhythmen schafften es bis in die Tagesthemen.

10
15
20

Vorläufiger Endpunkt dieser Entwicklung war die Behandlung des Richtlinienvorschlags im EU-Parlament Ende September 2003. Das Ergebnis dürfte allseits sowohl der Sache als auch dem Handwerklichen nach als mindestens ungut anzusehen sein. Die vom Parlament formulierten Änderungs-, Streichungs- und Ergänzungs-

25

⁶ www.eurolinux.org, siehe insbesondere dort auch die members-Liste, www.ffii.org, www.aful.org, <http://lpf.ai.mit.edu/Patents/>, etwas entfernter www.fsf.org, www.bustpatents.com und andere.

wünsche sind zahlreich, einige wenige seien in der Fußnote⁷ zitiert. Die dort zu findende Zusammenstellung mag eigenartig unzusammenhängend anmuten. Der Eindruck wird aber nicht besser, wenn man das Dokument als Ganzes betrachtet. „Jeder darf ‘mal‘, könnte das Motto gewesen sein, und so schaut das Ergebnis der Beratung aus. Nur Zyniker, die sich auf die unendlichen Stunden kostenträchtiger Anwaltsarbeit zur Trockenlegung des Sumpfes freuen, können sich wünschen, dass so etwas jemals in Kraft tritt.

1.11 Status heute: Rechtsgrundlagen

10

Es seien dem heutigen Status nach also kurz die derzeit geltenden Rechtsgrundlagen dargestellt:

Das EPÜ und das Deutsche Patentgesetz definieren gleichlautend den Erfindungsbegriff dahingehend, dass Software nicht darunterfällt. Das EPÜ sagt aber nichts dazu, was unter den Schutzbereich eines Patents fallen soll. Hierzu schweigt das EPÜ der Sache nach und verweist stattdessen mit Art.64 auf nationales Recht. Zum Schutzbereich sagt nationales Patentrecht (PatG § 9ff) *nicht*, dass Software oder softwareimplementierbare Erfindungen *nicht* in den Schutzbereich eines Patents fallen können. Unten bei 2.1 wird dies näher erläutert.

20

⁷ **Abänderung 8:** *Außerdem ist ein Algorithmus von Natur aus nichttechnischer Art und kann deshalb keine technische Erfindung darstellen.*

Abänderungen 16 u.a.: *Die Mitgliedstaaten stellen sicher, dass eine computerimplementierte Erfindung, die einen technischen Beitrag leistet, eine notwendige Voraussetzung einer erfinderischen Tätigkeit ist. Bei der Ermittlung des signifikanten Ausmaßes des technischen Beitrags wird beurteilt, inwieweit ...*

Abänderung 45: *Die Mitgliedstaaten stellen sicher, dass die Datenverarbeitung nicht als Gebiet der Technik im Sinne des Patentrechts betrachtet wird und dass Innovationen im Bereich der Datenverarbeitung nicht als Erfindungen im Sinne des Patentrechts betrachtet werden.*

Abänderungen 103 u.a.: *Die Mitgliedstaaten stellen sicher, dass die Erstellung, die Bearbeitung, die Verarbeitung, die Verbreitung und die Veröffentlichung von Informationen in jedweder Form niemals eine direkte oder indirekte Patentverletzung darstellen können, ...*

Abänderungen 104 u.a.: *Die Mitgliedstaaten stellen sicher, dass in allen Fällen, in denen in einem Patentanspruch Merkmale genannt sind, die die Verwendung eines Computerprogramms erfordern, eine gut funktionierende und gut dokumentierte Referenzimplementierung eines solchen Programms als Teil der Beschreibung ohne einschränkende Lizenzbedingungen veröffentlicht wird.*

Abänderung 107: *'Technischer Beitrag', auch 'Erfindung' genannt, ist ein Beitrag zum Stand der Technik auf einem Gebiet der Technik, der ...*

Das Urheberrecht gibt den Kopierschutz auf geschriebene Software. Es gibt aber ausdrücklich keinen Konzeptschutz.

- 5 TRIPS verpflichtet die Vertragsstaaten dazu, für den Patentschutz aller technischen Erfindungen zu sorgen. Eine Softwareausnahme ist hier weder vorgesehen noch für die Vertragsstaaten zugelassen.

1.12 Weiteres Verfahren

10

Weder die EPO noch nationale Regierungen werden vermutlich demnächst Anläufe unternehmen, irgendwelche Änderungen vorzunehmen.

- 15 Auf EU-Ebene wird die Richtlinie weiterbearbeitet werden, was auch das Ad-acta-
legen beinhalten kann. Ein Nachlassen der lobbyistischen Bemühungen der Gegner der Richtlinie ist jedenfalls nicht zu erwarten, eher das Gegenteil: Die bisherigen
Erfolge dahingehend, sowohl auf EPO-Ebene die Revision des EPÜ verhindert zu
haben als auch auf EU-Ebene über das Parlament gewaltigen Einfluss auf den Gang
der Dinge bekommen zu haben, legen die Vermutung nahe, dass dies eher beflü-
20 gelnd und anfeuernd denn sonstwie wirkt.

2. Rechtslage, Amtspraxis, Anmelderverhalten

2.1 Rechtslage

25

§ 1 des deutschen Patentgesetzes wie der oben zitierte gleichlautende Art. 52 des Europäischen Patentübereinkommens definieren den Begriff der Erfindung in der Weise, dass sie Software vom Katalog der Erfindungen, die die Erteilung eines Patents rechtfertigen können, ausschließen, siehe oben Fußnote 4. Dies ist ein im We-

sentlichen ins Leere laufender Ausschluss, denn er richtet sich gegen Software als Erfindung, nicht aber gegen Software als Implementierungsmittel von Erfindungen. Erfindungen in Software werden offensichtlich nicht gemacht und dann logischerweise auch nicht angemeldet. Sehr wohl werden aber Erfindungen gemacht, die mit Software „gebaut“ bzw. implementiert werden können, und die werden natürlich angemeldet – und sind vom Softwareausschluss nicht erfasst, weil Software nicht die Erfindung ist. Viele der softwareimplementierbaren Erfindungen können von Erfindern gemacht worden sein, die von Software keine Ahnung haben, wenn es überhaupt eine implementierende Software gibt. Dementsprechend findet man in dem, was manchmal im Jargon „Softwarepatent“ genannt wird, zwar alles mögliche an Inhalten, aber eines ausnahmslos nicht: Software. In keinem der sog. „Softwarepatente“ steht auch nur ein Fitzelchen Software. Da Software nie die Erfindung ist, läuft der Softwareausschluss in den Patentregelungen im Wesentlichen ins Leere.

Um einem Missverständnis vorzubeugen, sei hier nochmals darauf verwiesen, dass bei dieser Sichtweise der in den Gesetzestexten verwendete Terminus „Programme für Datenverarbeitungsanlagen ... als solche“ als Synonym zu „Software“ verstanden wird. Der Autor hat in den 15 Jahren seiner Berufspraxis noch keine Patentanmeldung gesehen, in der ein „Programm für eine Datenverarbeitungsanlage“ die Patentwürdigkeit hätte begründen sollen, und er hat nur eine einzige Patentanmeldung gesehen, in der zur Erläuterung der Erfindung ein Programm in der Programmiersprache C über ca. 30 Seiten abgedruckt war. Aber auch in dieser Anmeldung wurde das zitierte Programm nicht zur Begründung der Patentwürdigkeit herangezogen. Angesichts eines im Wesentlichen ins Leere laufenden Softwareausschlusses könnte man also achselzuckend alles beim Alten lassen. Ein ähnliches Verhalten findet man zu aufgelassenen Kaugummiautomaten an Hauswänden, deren Halbwertszeit sicher mindestens zehn Jahre beträgt. Der Softwareausschluss generiert aber neben seinem sachlichen Leerlauf Sekundärprobleme, die kurz stichwortartig aufgezählt seien:

- Historisch bedingte Grobklassifikation und damit schlechtere Recherchierbarkeit und Patentierungsergebnisse zu softwareimplementierbaren Erfindungen,
- 5 • jedenfalls früher Nichteinstellung von Informatikern als Prüfer in Patentämtern,
- Beeinflussung des Anmelderverhaltens wie unten bei 2.3 dargestellt, und
- weitere rechtlich motivierte Probleme, deren Darstellung hier zu sehr ins Detail gehen würde.

10

Zur Behebung dieser nachrangigen Auswirkungen wäre eine Korrektur dahingehend angebracht, den Softwareausschluss aus den Patentregelungen ersatzlos zu streichen. Im Großen würde sich nichts ändern, aber den obigen Sekundärproblemen wäre der Nährboden entzogen.

15

Der oben zitierte Art. 52 EPÜ und der gleichlautende § 1 PatG betreffen beide nicht den Schutzbereich eines Patents. Der Schutzbereich ist die Menge derjenigen gewerblichen Güter, die unter den Unterlassungsanspruch aus einem Patent fallen. Der Schutzbereich ist der Sache nach nicht im oben zitierten Art. 52 EPÜ oder in § 1 PatG geregelt, sondern in Art. 64 EPÜ, der aber auch „nur“ auf die nationalen Regelungen verweist, die im deutschen PatG die §§ 9ff sind. Eine Softwareausnahme gibt es hier nicht. Und ganz allgemein lauten das Gesetz und die Rechtsprechung seit Jahrzehnten eindeutig und ausnahmslos dahingehend, dass es bei der Verletzung einer patentierten Erfindung auf die konkreten Implementierung nicht an-

20 kommt, solange – gleich wie – alle Merkmale der Erfindung verwirklicht sind.

25

2.2 Amtspraxis

Die Praxis der Patentämter entspricht den geltenden Regelungen. Sie erteilen Patente auf softwareimplementierbare Erfindungen, solange nicht Software selbst die Erfindung ist (was praktisch niemals vorkommt) und natürlich soweit die Erfindung den übrigen Kriterien für die Patentfähigkeit genügt, hier insbesondere der Forderung nach technischem Charakter und nach erfinderischer Tätigkeit.

2.3 Anmelderverhalten

Interessant ist ein Blick auf das Anmelderverhalten: In den USA gibt es weder einen Softwareausschluss noch eine strikte Forderung nach technischem Charakter von patentierbaren Erfindungen. Dort wird „alles“ angemeldet, und arglos werden die meisten dieser Erfindungen genauso in Europa und in Japan eingereicht. Dort wird dann das Untechnische herausgefiltert (z. B. die „business method patents“), aber alles andere wird erteilt, soweit es den übrigen Kriterien für die Patentfähigkeit genügt. Am Rande vermerkt sei hier, dass in den USA, wo es keinen Softwareausschluss gibt, trotzdem keine Softwareerfindungen, also solche, wo Software die Erfindung wäre, angemeldet werden. Offensichtlich gibt es die nicht.

Die europäische Großindustrie verhält sich territorial gesehen andersherum, aber sonst ähnlich wie die US-Unternehmen: Aus solidem In-House-Patentwissen heraus meldet sie jedenfalls alles Technische in Deutschland bzw. sonstwo in Europa an, auch soweit es softwareimplementierbar ist, und springt damit auch in andere Kontinente und lässt allenfalls für die USA „business method patents“ links liegen, die sie dort ja auch erhalten könnte.

Bei kleinen und mittleren Unternehmen („KMU's“) ist dies, wie der Autor des Textes aus markanten eigenen Erfahrungen weiß, immer wieder anders: Viele haben irgendwo im Land gehört, dass „Softwarepatente illegal sind“, glauben dies mangels besseren eigenen Wissens auch und melden ihre softwareimplementierbaren

Erfindingen nicht an, weder zu Hause noch sonst wo auf dem Globus. Der Endpunkt dieser Entwicklung ist qualitativ eindeutig absehbar, man kann sich allenfalls um die Quantität des Effekts streiten: Jedenfalls was den Rechtsschutz ihrer softwareimplementierbaren Erfindungen angeht, geraten KMU's ins Hintertreffen gegenüber der Großindustrie und gegenüber der sachlich vergleichbaren überseeischen Konkurrenz, egal wie groß oder klein diese Unternehmen sind.

3. Argumente, Paradigmen, Sichtweisen

Die ökonomische Rechtfertigung von Patenten als zeitlich begrenztes Verbotungsrecht zum Schutz tatsächlicher Innovation liegt darin, dass durch sie vor Beginn von Entwicklungsarbeiten die Gewissheit gegeben ist, dass tatsächliche Erfindungen dann auch eine begrenzte Zeit in der Sphäre des entwickelnden Unternehmens bleiben, sodass vorab ein Anreiz für die Entwicklung gesetzt wird, und dass nach erfolgreichem Abschluss der Entwicklungsarbeiten die berechtigte Hoffnung auf Amortisation und auch Überamortisation von Entwicklungskosten besteht, was man als Belohnung für die Entwicklung ansehen kann.

Nachfolgend sollen in den Punkten 3.1 bis 3.15 diejenigen Argumente, Paradigmen, Sichtweisen und Insinuationen dargelegt werden, die seitens der Gegner von Patenten auf softwareimplementierbare Erfindungen zur Unterstützung ihrer Forderungen und Wünsche verwendet werden. Die Tatsache, dass die „Argumente“ hier aufgeführt werden, heißt nicht, dass der Autor ihnen Richtigkeit beimessen würde. Sie werden hier genannt um zu zeigen, wie argumentiert wird, um abschätzen zu können, in welcher Weise und auf welcher Ebene Beeinflussungen stattfinden und mit was auch zukünftig zu rechnen ist.

3.1 „Patente passen auf Software nicht, denn Software ist etwas Besonders, weil ...“

... ja, weil ... damit in Europa 400.000 Leute befasst seien, ... Software Abbild menschlichen Denkens sei, ... Software und Sprache und Denken und Kreativität dasselbe seien, ... Software ein Kulturgut geworden sei, ... Software sehr viele Schnittstellen und Standards habe, die nicht patentiert werden dürften, ... Softwarefirmen Basisökonomie kleiner Firmen seien, ... die Programmierer sonst Patentrecherchen machen müssten, ... Softwarepatente zu Überamortisationen führen könnten, ... Patente auf Software immer wieder die einzige mögliche Lösung blockieren würden, ... Software im Internet ganz einfach verteilt werden könne, ... Software eine ganz andere Vertriebs- und Kostenstruktur habe, ... Software ganz außergewöhnlich innovativ und deshalb viele Jahre Patentschutz hinderlich seien, ... Software das Open-Source-Geschäftsmodell erlaube, usw..

Bei vielen dieser Statements lässt sich das Wort „Software“ schlicht durch „Hardware“ oder durch „Eisen“ ersetzen, und das Statement ist genauso richtig oder halb-richtig wie vorher. Dies ist ein Indiz dafür, dass die Aussage an sich zwar (halb-)richtig sein mag, es sich dabei aber eben nicht um eine Softwarebesonderheit handelt. Und auch das selbstverständlich unterstellte Paradigma „Unterschiedliches muss unterschiedlich behandelt werden“ sei in Abrede gestellt: Ein Fahrrad ist sicher etwas anderes als ein Lastkraftwagen. Aus guten Gründen aber gilt eine Ampel für beide in gleicher Weise.

Wenn auf mögliche „Überamortisationen“ (heißt: wegen des Patentschutzes erhält der Inhaber mehr, als er in seine Erfindung investieren musste) hingewiesen und dies als verwerflich charakterisiert wird, sei daran erinnert, dass es Ziel jedweder Investition und allgemein jedweden Wirtschaftens ist, mehr herauszuholen, als man hineinsteckt. Nicht Nullsummenspiel oder das Draufzahlen stimulieren, sondern Gewinnerwartung. Wenn diese Zielsetzung diffamiert wird, dann wird die Mehrwertschaffung insgesamt in Frage gestellt.

Wenn die einfache Verteilbarkeit von Software im Internet als patentunzuträgliche Besonderheit bemüht wird, wird – patentrechtlich betrachtet – die faktische Einfachheit der Rechtsverletzung als Begründung zur Abschaffung des Rechts verwendet. Man möge sich vorstellen, was analog geschehen würde, wenn der Chirurgenverband forderte, Chirurgen seien von Kunstfehlerhaftung freizustellen, weil ein Kunstfehler sehr leicht vorkommen könne.

Die unübertroffene Innovationskraft von Software wird selbstverständlich behauptet, aber in seiner Pauschalität sei das Statement hier trotzdem in Frage gestellt. Auch Hardware ist außerordentlich innovativ, was jeder daran feststellen kann, dass er sich spätestens alle vier oder fünf Jahre nicht nur neue Software kaufen muss, sondern eben auch neue Hardware. Abgesehen davon ist Software nicht zwingend immer außerordentlich schnell und innovativ. Das Paradebeispiel für nicht innovative Software war das Jahr-2000-Problem Ende 1999, verursacht durch immer noch laufende Altsoftware, die vierstellige Jahresangaben nicht handhaben konnte. Ein anderes Beispiel: Ein bekanntes Betriebssystem verwendete über weit mehr als ein Jahrzehnt hinweg zuletzt jedenfalls noch intern konstante Formate, beispielsweise das 8.3-Dateinamensformat, obwohl es längst obsolet geworden war.

Und im Übrigen vernachlässigen die Verweise auf vermeintliche Softwarebesonderheiten die Tatsache, dass Software ja regelmäßig nicht die Erfindung ist. Software wird nur zum „Bau“ der Erfindung benötigt, so wie man Eisen zum Bau eines erfundenen Zahnrades braucht, siehe auch oben 2.1. Es gehen dann Verweise auf Softwarebesonderheiten der Erfindungen schon dem betrachteten Gegenstand nach ins Leere, man redet an der Sache, nämlich an den real vorliegenden Erfindungen, vorbei.

3.2 „Urheberrecht reicht ...“

... für Software, wird behauptet, und Patentschutz für softwareimplementierbare Erfindungen bräuchte man deshalb nicht. Soweit Software alleine zu betrachten ist, wird man dem obigen Statement sicher zustimmen können. Nur ist es eben so, dass
 5 das Patentrecht erfinderische Konzepte schützen will und diese Erfindungen regelmäßig – wie schon gesagt – nicht Software sind. Zur Vermeidung von Wiederholungen wird insoweit auf das Ende des vorherigen Punktes verwiesen. Zum Schutz von Konzepten ist das Urheberrecht aber völlig untauglich.

10 3.3 „Software ist untechnisch“

Schon ganz vorab wird man vermuten können, dass das Statement genauso richtig ist wie die Aussage „Alle Iren haben rote Haare“. Es mag ja durchaus sein, dass softwareimplementierbare Erfindungen mit untechnischen Gegenständen stärker
 15 korrelieren als metallimplementierbare Erfindungen dies tun. Dies ändert aber nichts daran, dass die Fragen nach Technizität einerseits und nach Software andererseits voneinander unabhängig und jeweils in Ansehen einer konkreten Erfindung getrennt voneinander zu beantworten sind. Und abermals gilt das Gleiche wie am Ende von 3.1 und 3.2 gesagt: Auch die Frage der Technizität ist nicht in Ansehen
 20 einer Software zu beurteilen, sondern in Ansehen einer Erfindung, die regelmäßig jenseits der implementierenden Software liegt. Die Frage nach Technizität von Software ist im patentrechtlichen Kontext somit eigentlich irrelevant. Ob sich in anderen Sphären jemand dafür interessiert, weiß der Autor nicht.

25 3.4 „30.000 illegale Swpat“ (Swpat = „Softwarepatent“)

Irgendwie kam die Zahl in die Welt, dass das EPA bisher 30.000 Patente auf softwareimplementierbare Erfindungen erteilt habe. Aus der Tatsache, dass Software ein Patent verletzen kann, wird angesichts des Softwareausschlusses in Art. 52 EPÜ

hergeleitet, dass diese Patente illegal seien. Selbstverständlich wird bei diesem Argument vorausgesetzt, dass Art. 52 auf den Schutzbereich eines Patents zu lesen ist, obwohl Art. 52 doch den Erfindungsbegriff definiert und demgegenüber der Schutzbereich ganz woanders, nämlich in Art. 64 EPÜ⁸ (Verweis) bzw.

5 §§ 9ff. PatG definiert ist. Eine Softwareausnahme ist dort nicht gemacht. Gleichwohl wird ein Prima-facie-Widerspruch dazu benützt, die Amtspraxis von EPA und DPMA als contra legem darzustellen. Nach Auffassung des Autors verfängt dieses „Argument“ in hohem Maße und trägt stark zu Skepsis und Vertrauensverlust gegenüber der bisherigen Handhabung und den Akteuren bei.

10

3.5 Ängste schüren

Das hört sich so an: „Mozart hätte Stehgeiger werden müssen, wenn Haydn ein Patent auf seine Musik bekommen hätte“. „Softwareunternehmen werden in einen Rüstungswettlauf gedrängt, bald haben wir amerikanische Verhältnisse“. „Siehe patenting-art.com“. Das Verfügungsverfahren von amazon.com gegen Barnes & Noble in der Weihnachtszeit 2000 (?) aus dem One-klick-Patent heraus war ein gefundenes Fressen insoweit. Genüsslich wird Buch geführt darüber, wie sich in Amerika welche Unternehmen mit Patentrechtsklagen überziehen. Routiniert und zuverlässig werden diese Aspekte vorgebracht, angeführt und immer wieder betont, und trotz des offenkundigen Widersinns wird dies gemäß der Strategie „aliquid semper haeret“ sicher in bestimmtem Umfang verfangen.

20

3.6 Vermengte Betrachtung

⁸ **EPÜ Art. 64:** Rechte aus dem europäischen Patent

(1) Das europäische Patent gewährt seinem Inhaber von dem Tag der Bekanntmachung des Hinweises auf seine Erteilung an in jedem Vertragsstaat, für den es erteilt ist, vorbehaltlich Absatz 2 dieselben Rechte, die ihm ein in diesem Staat erteiltes nationales Patent gewähren würde.

(2) Ist Gegenstand des europäischen Patents ein Verfahren, so erstreckt sich der Schutz auch auf die durch das Verfahren unmittelbar hergestellten Erzeugnisse.

(3) Eine Verletzung des europäischen Patents wird nach nationalem Recht behandelt.

Bei der Argumentation gegen die Patentierung softwareimplementierbarer Erfindungen kommt es regelmäßig dazu, dass die unterschiedlichsten Aspekte in einen Topf geworfen und zusammen betrachtet werden. In einem Atemzug erfolgt die

5 Betrachtung von Erfindung, Software, Implementierung, Klarheit, Schutzbereich, Technizität und Trivialität, und wenn in diesem Gebräu irgendetwas eigenartig erscheint, wird dies zur Begründung dafür, dass Patente auf softwareimplementierbare Erfindungen unpassend seien, verwendet. Auf diese Weise entstehen unterschiedliche Permutationen von doch eigentlich immer wieder den gleichen „Argumenten“.

10 Unterschiedliche Rechtsfragen werden vermengt dargestellt, die in handwerklich ordentlicher Weise eigentlich getrennt behandelt werden müssten und dann auch übersichtlich werden würden.

3.7 Übergehen der sonstigen Ausschlusskriterien

15

Es wird so getan, als ob der Softwareausschluss des Art. 52 EPÜ „die letzte Barriere vor der grenzenlosen Patentierbarkeit“ sei, und wenn die fiele, dann sei der Patentierung der menschlichen Denke keine Grenze mehr gesetzt. Übersehen oder absichtlich übergangen wird dabei zweierlei, nämlich erstens, dass der existierende

20 Softwareausschluss faktisch ins Leere läuft und somit keine „Filterwirkung“ hat, und zweitens, dass es wesentlich mächtigere Schranken als den Softwareausschluss gibt, nämlich insbesondere die Beschränkung des Patentfähigen auf Technisches und auf Erfinderisches. Diese Aspekte gehen regelmäßig unter oder werden verschwiegen, und stattdessen wird der schale Eindruck vermittelt, dass es gewisse

25 Kreise gäbe, deren Ziel die Patentierung von allem und jedem sei.

3.8 Verschwörungstheorien

Denjenigen, die zugunsten eines vernünftigen Patentschutzes auch für softwareimp-

lementierbare Erfindungen reden, werden entweder mangelndes Verständnis oder finstere eigene Interessen unterstellt. Patentanwälte wollen sich mit der EPÜ-Revision bzw. mit der EU-Richtlinie angeblich ein neues Geschäftsfeld erschließen, Amerika will Europa patentrechtlich so usurpieren, wie es das mit Japan schon geschafft habe, usw..

An sich sind diese Darstellungsarten subsidiär zu der unter 3.4 dargestellten vermeintlichen Unrechtmäßigkeit von Patenten auf softwareimplementierbare Erfindungen, denn die dargestellten Szenarien „funktionieren“ ja nur, wenn es tatsächlich richtig wäre, dass Patente auf softwareimplementierbare Erfindungen bisher unzulässig seien, was aber ja nicht der Fall ist. Zur Diskreditierung des Standpunktes zugunsten eines vernünftigen Patentschutzes taugen solche Szenarien aber allemal.

3.9 „270.000 Stimmen, 2.000 Firmen gegen Logikpatente“

Die Richtliniengegner berufen sich auf eine hohe Zahl von Unterstützern der eigenen Position. Dies wird regelmäßig angeführt und zur Legitimation der eigenen Forderungen verwendet. Verifizierbar sind diese Zahlen nicht. Verwendet werden sie aber gleichwohl, und es mag sein, dass die hohen Zahlen durchaus den einen oder anderen Parlamentarier beeindrucken.

Beeindruckend in die entgegengesetzte Richtung sollten dann aber zumindest auch die angeblich 30.000 erteilten Patente auf softwareimplementierbare Erfindungen sein. Sieht man sie als „Abstimmung mit den Füßen“ an, dann ist den genannten hohen Zahlen eine durchaus gewichtige Zahl entgegenzusetzen, denn diese 30.000 „Meinungsäußerungen“ sind nicht durch billig abzuschickende E-Mails oder effizient auf Messeständen (Cebit, Systems) in der Open-Source-Halle eingesammelte Unterschriften zustande gekommen, sondern durch ein viele Tausend Euro kostendes Verhalten. Und vermerkt sei hier auch, dass sich große IT-Verbände, z. B. in

Deutschland die Gesellschaft für Informatik (GI), für einen vernünftigen Patentschutz und für eine Einordnung der Softwareindustrie in herkömmliche ingenieurmäßige Sichtweisen aussprechen.

5 3.10 Mythos KMU

„KMU“ steht für „kleine und mittlere Unternehmen“. Seitens der Gegner von Patenten auf softwareimplementierbare Erfindungen wird angeführt, dass sich Patente angeblich besonders gegen kleine und mittlere Unternehmen auswirken würden.

- 10 Hier will man mit der Davids-Position im David-gegen-Goliath-Szenario Sympathien sammeln. Nach Auffassung des Autors ist es aber nüchtern betrachtet gerade anders herum richtig: Lässt man sich auf die Sichtweise David-gegen-Goliath ein, so hätte ein innovativer David mit einem oder mehreren Patenten in der Hand gegen
- 15 Goliath die Patente aus dem Repertoire streicht. Dann bliebe es rein auf dem Niveau des Armdrückens am Markt, und da dürfte Goliath in der Regel die wesentlich besseren Chancen, weil in jedem Fall den längeren Atem, haben.

- Im Übrigen unterscheiden die Gegner der Richtlinie durchaus zwischen „guten“
- 20 KMUs und „unguten“ KMUs – was zeigt, dass es um die Interessen von KMU's als solchen eigentlich gar nicht geht. Die „Guten“ sind diejenigen, die selber programmieren, Software schreiben, vielleicht auch innovativ sind, aber nichts patentieren. Die „Unguten“ sind diejenigen, die zwar auch kleine Unternehmen sind, softwareimplementierbare Erfindungen machen, diese aber patentieren lassen und dann Li-
- 25 zenzen einfordern. Wenn solch ein KMU als David gegen Goliath antritt und vielleicht auch noch einen Erfolg errungen hat⁹, wird das nicht als ein dem Patentsystem zu verdankender Erfolg eines KMU gesehen, sondern das KMU ist ein ungutes KMU, und die Sympathien liegen zwar nicht auf Seiten Goliaths, aber man hat doch

⁹ z. B. Eolas, siehe eolas.com

viel Einfühlungsvermögen dafür, dass Goliath sich über solche wadelbeißenen KMUs ärgert ... wobei, um die Sache rund zu machen, dieser Ärger nicht auf Goliaths Rechtsverletzung zurückgeführt wird, sondern auf „Softwarepatente“, verbunden mit dem Hinweis, dass demnächst sich wohl auch KMUs in dieser Weise über

5 „Softwarepatente“ würden ärgern müssen ...

Der Autor hat Software-Kleinunternehmen als Mandanten, deren eine zeitkonsumierende Tätigkeit es ist, Geld von Investoren aufzutreiben. Regelmäßig fordern diese, das im Betrieb entstehende „Intellectual Property“, also auch die software-

10 implementierbaren Erfindungen, in vernünftigem Umfang abzusichern – und so geschieht es dann auch. Wenn das nicht möglich wäre, wären Investoren zurückhaltender, denn warum sollten Sie Geld in ein Unternehmen stecken, dessen erfinderische Entwicklungen (weil sie das „Pech“ haben, softwareimplementierbar anstatt z. B. metallimplementierbar zu sein) nicht gegen Übernahme durch die Konkurrenz

15 geschützt werden könnten? Nun, so ist es ja nicht, denn softwareimplementierbare Erfindungen sind nicht a priori vom Patentschutz ausgeschlossen, siehe oben 2.1. Aber wenn es so wäre, dann würde sich dies auch im Hinblick auf die Kapitalversorgung von Kleinunternehmen gegen KMU's auswirken. Man tut also auch unter diesem Aspekt den Software-KMU's keinen Gefallen, wenn man sich pauschal ge-

20 gen Patente auf softwareimplementierbare Erfindungen ausspricht.

3.11 „Ich wandere aus“

Es wird eine Situation beschworen, in der man vor lauten Patenten angeblich nicht mehr programmieren könne. Pathetisch wird dann der Arbeitsplatzverlust angedroht

25 und das Auswandern als die einzig mögliche Alternative beschworen. Abgesehen davon, dass man damit dem sich territorial erstreckenden Schutz von Patenten nicht entkommt und weiter abgesehen davon, dass sich der Patentschutz ja gerade nicht auf Forschung und Entwicklung erstreckt, ist es dann umso erstaunlicher, dass man

bei der Frage an ein Auditorium von vielen Hundert Rezipienten dahingehend, wer denn schon einmal wegen eines Patentes etwas unterlassen haben müsse oder wer auch nur ein einziges Mal wenigstens vorsichtshalber eine Patentrecherche gemacht habe, eine Null-Antwort bekommt. Man erhält dann nur Antworten dahingehend,
 5 dass man sich Sorgen mache und dass man ja wohl (insbesondere amerikanischen Verhältnissen) vorbeugend tätig werden dürfe.

3.12 Konsequenzen in Europa wegen amerikanischer Verhältnisse

10 Mit großer Selbstverständlichkeit werden seitens der Richtliniengegner tatsächliche oder vermeintliche Verhältnisse in Amerika zur Begründung ihrer Anliegen in Europa verwendet. Dies geschieht entweder unmittelbar, oder etwas ausgefeilter im Hinblick auf Prävention amerikanischer Verhältnisse. Der Ansatz ist aber nach Auf-
 fassung des Autors insoweit falsch, als „amerikanische Verhältnisse“, soweit sie
 15 denn tatsächlich bedenklich sein sollten, sicher nicht in der Tatsache liegen, dass das US-Recht keinen Software-Ausschluss in seinen Patentregelungen hat. Wenn man dem amerikanischen Rechtssystem skeptisch gegenübersteht, dann ist das häufig aus der zu beobachtenden Prozessfreude¹⁰ und gerne berichteten – realen? – Absurditäten herzuleiten, hat aber mit den Softwareregeln nichts zu tun. Insofern
 20 sind dann auch Schlussfolgerungen zu europäischen Softwarediskussionen abwegig.

3.13 „Linux hätte nicht entstehen können, wenn es Softwarepatente gegeben hätte“

25 Das „Argument“ ist grundfalsch, denn seine Prämisse ist falsch: Es gibt Patente auf softwareimplementierbare Erfindungen, angeblich sogar 30.000 davon in Europa (siehe 3.4), und Linux und andere Open-Source-Produkte existieren trotzdem sehr unbeschwert. Und nicht nur existieren sie in Europa, sondern auch in den USA, die

¹⁰ Zitat US-Anwalt: „I like Clausewitz: Lawyering is continuation of business with other means“

den angeblichen „Sündenfall“ zu Softwarepatenten schon hinter sich haben. Für Europa wird der Widerspruch quasi-juristisch klingend, aber fachlich völlig falsch damit erklärt, dass die Rechtslage, insbesondere der Softwareausschluss in Art. 52 EPÜ, die Durchsetzbarkeit der fraglichen Patente unmöglich mache, so dass die Patentinhaber dieser Tag (noch) alle kollektiv stillhielten. Eine Antwort auf die Frage, wie sich der Widerspruch für die USA löse, wo es ja keinen Softwareausschluss entsprechend Art. 52 gibt, erhält man nicht mehr.

3.14 Reduktion von Patenten auf Verbotungsrechte

10

Seitens der Gegner der Richtlinie wird die durch das Patentrecht vorgenommene Interessensgestaltung darauf reduziert, dass ausschließlich der Verbotungsaspekt dargestellt wird. Sicherlich ist dieser vorhanden, und er wird durchaus gelegentlich Unternehmen in ihren gewerblichen Vorhaben behindern. Patente haben aber auch Anreiz-, Belohnungs- und Sicherheitsaspekte und damit eine markante entwicklungsfördernde Wirkung, wie dies vor 3.1 weiter oben dargestellt wurde. Diese Aspekte werden regelmäßig nicht erwähnt, sie werden schlicht übergangen.

15

3.15 „Softwarepatente sind trivial“

20

Die Nennung dieses Arguments erfolgt mit Absicht am Ende der bisherigen Aufzählung, denn der Autor misst ihm noch am ehesten einen Kern möglicher Relevanz bei. In jedwedem der oben angeführten Kontexte wird ausdrücklich oder implizit deutlich, dass man den Gegenstand von Patenten auf softwareimplementierbare Erfindungen jedenfalls in den allermeisten Fällen für trivial und das Patent deshalb für zu Unrecht erteilt hält. Diese Sichtweise schürt einen wesentlichen Teil der Empörung gegen Patente auf softwareimplementierbare Erfindungen.

25

Ein Teil dieser Auffassung ist sicherlich Ex-post-Betrachtungen zuzuschreiben. In Ansehen eines bestimmten Patents wird dessen Gehalt ausgehend von heutigem Wissen als trivial angesehen. Gelegentlich handelt es sich aber bei den betrachteten Patenten um US-Patente mit frühen Anmeldetagen, die wegen der ehemals anderen Laufzeitbestimmungen in den USA heute noch in Kraft sind – nicht aber in Europa. Damit lassen sich aber nicht alle Trivialitätswahrnehmungen erklären. Klassischer Fall hier war abermals das One-klick-Patent von Amazon (US 5 960 411), das – verkürzt dargestellt - den Gedanken schützt, dass man bei Käufen im Internet ausgehend von einer Produktdarstellung auf dem Bildschirm einen Kauf durch lediglich einen einzigen Mausklick bewirken kann anstatt mehrerer, wie es vorher üblich war. Das zitierte One-klick-Patent ist aber beileibe nicht das einzige, das Bauchgrimmen verursacht. Fast jedes Patent wird als trivial bezeichnet, und man kann sogar Bemühungen dahingehend beobachten, für softwareimplementierbare Erfindungen eine inhärente, systemnotwendige Trivialität zu begründen.

15

In seinen blumigen Auswüchsen kann man diese Argumentationslinie sicherlich nicht ernst nehmen, zumal sie sich ja offensichtlich mit dem auch sehr routiniert vorgetragenen, weiter oben bei 3.1 schon gestreiften Statement beißt, wonach die Softwarebranche angeblich ganz außerordentlich innovativ sei, was sich mit notorischer Trivialität der dabei entstehenden Erfindungen nicht gut verträgt.

20

Gleichwohl aber soll hier gefragt werden, ob der Vorwurf der Trivialität nicht einen wunden Punkt berührt. Wenn man angesichts des Vorwurfs auf die Bestimmungen im Patentgesetz dahingehend, dass erfinderische Tätigkeit eine der Patentfähigkeitsvoraussetzungen sei, hinweist, erntet man Hohngelächter, verbunden mit der Feststellung, dass das Kriterium ausgedient habe und es letztendlich nicht oder nicht ordentlich angewendet werde.

25

Nach Auffassung des Autors kann letzteres nicht in jedem Fall verneint werden.

Wenn die Richtliniengegner sich als in einem „Gestrüpp von Trivialpatenten stehend“ darstellen, dann ist dies sicherlich Rhetorik und übertrieben. Wenn aber festgestellt wird, dass durchaus auch Banales oder Obskures patentiert wird, dann ist dies nicht in jedem Fall zu verneinen – im Übrigen auch auf Feldern jenseits softwareimplementierbarer Erfindungen. An dieser Stelle ist nach Auffassung des Autors durchaus Aufmerksamkeit angebracht: Es dürfte Common Sense sein, dass die Patentierung von Trivialitäten nicht erwünscht ist, denn ein Filz von Patenten auf einfach abgewandelte Gegenstände würde nicht mehr unter die Rechtfertigung des Patentschutzes im Allgemeinen fallen, ja würde sogar das Gegenteil bewirken, nämlich ökonomisch nicht wünschenswerte und ungerechtfertigte Behinderungen und eine Verrechtlichung des wirtschaftlichen Bereichs.

Auf der Jahrestagung der „American Intellectual Property Law Association“ (AIPLA) Ende Oktober 2003 redete der Vorsitzende der „Federal Trade Commission“ (FTC), T. J. Muris, sehr direkt zu genau dieser Problematik, die er mit „questionable patents“ umschrieb¹¹. Seitens der FTC werden Abhilfen als notwendig angesehen, und insoweit machte sie auch bestimmte Vorschläge¹², die man aber tendenziell als eher verfahrenstechnisch charakterisieren kann. Gleichwohl ist festzuhalten, dass in den USA Regierungsstellen insoweit Handlungsbedarf sehen und Vorstöße machen.

4. Taktik

In zunehmendem Maße ist festzustellen, dass taktische Finessen und Finten die sachliche Argumentation verdrängen. Dies gilt für beide Seiten in der Diskussion.

Seitens der Gegner der Richtlinie ist zu bemerken, dass mit langem Atem intensiver, handwerklich gut gemachter Lobbyismus jeweils vor den richtigen Stellen im

¹¹ siehe <http://www.ftc.gov/opa/2003/10/murisaipla.htm>

richtigen Timing betrieben wird. Schon das, was davon öffentlich sichtbar ist, fordert Zeit, Know-how und Geld. Nach nicht belegbarer Vermutung des Autors werden eine oder mehrere Personen hierfür entweder bezahlt oder in ihrer Anstellung bei interessierten Firmen insoweit freigestellt. Kaum vorstellbar erscheint, dass der
5 Aufwand ausschließlich „nach Feierabend“ erbracht wird.

Und auch seitens der Befürworter eines vernünftigen Patentschutzes sind Taktierereien erkennbar. Die EU-Kommission führt das Sujet unter dem Titel „Computerimplementierbare Erfindungen“, ganz so, als ob der Computer das patentrechtliche
10 Problem aufgeworfen hätte oder erfindungsgemäß individualisiert sei. Ganz klar ist dies aber nicht der Fall: Weder sind Computer bei softwareimplementierbaren Erfindungen erfindungsgemäß individualisiert, noch werfen Computer patentrechtliche Probleme auf. Vielmehr ist es die Software, die als Implementierung der Erfindung dementsprechend individualisiert ist, und der Softwareausschluss in den Pa-
15 tentregelungen ist es, der patentrechtliche Probleme aufwirft. Wohl aber in der Hoffnung, die Richtlinie unauffällig durchwinken zu können, wurde ein unverfänglicher Wortlaut gewählt. Einige der Sachfragen blieben dabei allerdings auf der Strecke. Auffällig war beispielsweise, dass eigenartigerweise der ursprüngliche Richtlinienentwurf der Kommission sich kaum mit dem aus dem Softwareaus-
20 schluss von EPÜ Art. 52 entspringenden Problem befasste. Vielmehr wurde ausführlich zur Technizität softwareimplementierbarer Erfindungen Stellung genommen, was zwar durchaus auch ein interessantes Problem ist, aber eben sachlich gesehen neben dem Softwareausschluss von Art. 52 EPÜ steht.

25 Ergebnis all dessen sind nun der schon vor Jahren gescheiterte Versuch des EPÜ, die entsprechenden Regelungen anzupassen, und ein vorläufig verunglücktes Richtlinienprojekt der EU, Ausgang offen.

¹² siehe <http://www.ftc.gov/os/2003/10/innovationrpt.pdf>

5. Interessenlagen

Hilfreich mag ein Blick auf die unterschiedlichen Interessenlagen sein:

- 5 Die Gegner von Patenten auf softwareimplementierbare Erfindungen wünschen sich letztendlich, dass ihr „Werkstoff“, nämlich Software, frei von patentrechtlichen Eingriffen ist. Der Wunsch speist sich sicher auch aus dem Open-Source-Gedanken heraus und lässt sich in etwa so formulieren: „Open-Source ist ein vergleichsweise einmaliges Entwicklungsmodell für Software. Es lebt nicht nur, aber auch vom Idealismus vieler Programmierer. Schade wäre es, wenn dies durch Patente unterbunden würde“. Daneben gibt es aber sicher auch den nicht idealistisch, sondern im banalen Eigeninteresse formulierten Wunsch, von Rechten Dritter nicht behelligt zu werden.
- 10
- 15 Zugegebenermaßen ist der der Open-Source-Idee entspringende Gedanke nicht unsympathisch, und gerade in seinem sichtbarsten Ergebnis, Linux als Alternative zu Windows, wünscht man sich, dass diese Schiene Erfolg hat. Nur ist es eben nicht richtig, die Unvereinbarkeit von Open-Source und Patenten auf softwareimplementierbare Erfindungen herbeireden zu wollen. Der beste Beweis für das Gegenteil, also für die Vereinbarkeit von Patentschutz auch für softwareimplementierbare Erfindungen und Open-Source, ist doch die Realität selbst: Linux und andere Open-Source-Produkte existieren, obwohl (angeblich) 30.000 „Softwarepatente“ existieren. Man hat bisher nicht gehört, dass Linux irgendwelche Schwierigkeiten erfahren hätte. Und erstaunlicherweise ist dies nicht nur in Europa so, sondern auch im angeblichen Hexenkessel USA, wo alles noch viel schlimmer als in Europa sein soll.
- 20
- 25 Auch dort erfreuen sich insbesondere Linux und allgemein Open-Source eines fröhlichen, von Patenten ungestörten Lebens, siehe insoweit auch 3.13.

Linux ist sowohl in Europa als auch in den USA längst gesellschaftsfähig geworden. Rechtlich gesehen ist es so unbehindert, dass auch große Firmen wie IBM beschlossen haben, diese Schiene zu unterstützen und sich mit diesem Image zu verbinden. Langer Rede kurzer Sinn: Nach Auffassung des Autors existiert die insinu-

5 ierte Unvereinbarkeit von Open-Source einerseits und Patenten auf softwareimplementierbare Erfindungen andererseits nicht. Die Realität sowohl in Europa als auch in den USA zeigt, dass die Systeme sich miteinander vertragen. Das aus Verbrauchersicht durchaus wünschenswerte Schöpfungs- und Geschäftsmodell Open-Source ist an sich kein Grund, die Patentfreiheit für softwareimplementierbare Er-

10 findungen zu fordern, denn dieser Tage leben Open-Source und das Patentsystem sowohl in Europa als auch in den USA weitgehend in friedlicher Koexistenz.

Einen vernünftigen Patentschutz unter den wohlverstandenen Patentfähigkeitskriterien – hier insbesondere Technizität und erfinderische Tätigkeit – dürfte die Mehrheit der wirtschaftlich Tätigen bejahen. Auch haben sich in Deutschland die großen

15 Dachverbände, in deren Kompetenz Software fällt, – allen voran die Gesellschaft für Informatik (GI) – dafür ausgesprochen. Dies entspricht der Auffassung, dass der „Werkstoff“ Software in zunehmender Weise in normalen industriell-ingenieursmäßigen Kontexten steht und dass sich deshalb die gleichen Interessenlagen insbesondere hinsichtlich des Schutzes tatsächlicher Erfindungen ergeben.

20

Niemand oder nur Zyniker werden sich die Patentfähigkeit von allem und jedem wünschen. Damit würde das System ad absurdum geführt. Wenn sich die tatsächliche Systemhandhabung jenseits der ökonomischen Rechtfertigung des Systems bewegt, dann würde dies zu Akzeptanz- und Sinnverlust führen und so zu einem

25 Standortnachteil für das davon erfasste Territorium werden.

6. To do

Ist die Stellungnahme des EU-Parlaments zum Richtlinienvorschlag der Kommission eine Katastrophe? Oder wird sie irgendwann als Zwischenfall angesehen werden? Der Autor weiß es nicht. Da aber aus heutiger Sicht zumindest nicht sicher ist, dass sich schon alles von selbst vernünftig regeln wird, folgt unten forsch eine To-

5 Do-Liste:

6.1 Die EU-Richtlinie

Was den Entwurf der EU-Richtlinie angeht, ist das, was aus dem Europäischen Par-
 10 lament gekommen ist, in der vorliegenden Form unbrauchbar. Wenn eines Tages
 eine sinnvolle EU-Richtlinie existieren soll, wird viel Sach- und Überzeugungsar-
 beit notwendig, um diese herbeizuführen. Man könnte hier angesichts der mahlen-
 den Mühlsteine durchaus skeptisch werden. Offensichtlich sind die Befürworter
 eines vernünftigen Patentschutzes weitaus schlechter organisiert als die Richtlinien-
 15 gegner, was letztere sicher dahingehend abgeändert formulieren würden, dass die
 Patentbefürworter ihren Einfluss eher nicht-öffentlich in Expertengremien und auf
 der Ebene der Exekutive gelten gemacht hätten, während die Gegner sich eher ba-
 sisdemokratisch an die Öffentlichkeit und die Volksvertreter der Legislative wen-
 den würden, die es dann auch „gerichtet“ hätten.

20

Wohin dieses „Armdrücken“ führen wird, ist nach Kenntnis und Auffassung des
 Autors derzeit nicht absehbar. Seitens der Richtliniengegner ist man aber optimis-
 tisch: Auf die verunsicherte Frage „*Eine komplette Ablehnung des Ganzen* [d. h. der
 Richtlinie – Anm. d. Autors] *wäre doch in unserem Sinne, oder?*“ mailte der Spiri-
 25 tus Rector, ein ausgebildeter Linguist, am 24. September 2003 unmittelbar nach der
 Entscheidung des EU-Parlaments: „*Nein. Die RiLi ist jetzt unser Projekt. Die Rollen
 haben sich vertauscht*“¹³.

¹³ siehe <http://lists.ffii.org/archive/emails/swpat/2003/Sep/0723.html>

6.2 Technizität

Es ist Auffassung des Autors, dass Fragen nach der Technizität einer Erfindung bei softwareimplementierbaren Erfindungen durchaus häufiger kritisch sein mögen als bspw. bei Erfindungen, die zu ihrem Bau z. B. Metall brauchen. Technizitätsprobleme korrelieren mit der Softwareimplementierbarkeit einer Erfindung möglicherweise so wie die rote Haarfarbe mit den Personen irischer Herkunft. Wenngleich es dann de jure gerechtfertigt wäre, angesichts des anstehenden Software-Diskurses zur Frage der Technizität von Erfindungen „Andere Baustelle!“ zu rufen, wäre es doch hilfreich, wenn man auch in Sachen Technizität ein Stück vorankäme.

Die Frage der Technizität betreffend ist in den letzten Jahren ein systematischer Ansatz – ähnlich dem BGH-Urteil „Rote Taube“ – nicht erkennbar, und ein solcher Ansatz ist auch jenseits der (quasi-)judikativen Spruchkörper schwer zu leisten, denn zuletzt müssen ja eben die Spruchkörper herausarbeiten, was als technisch anzusehen ist und was nicht – ganz abgesehen davon, dass es auch sachlich sehr schwierig sein dürfte, eine überzeugende Definition oder auch nur einen guten Kriterienkatalog zu entwickeln.

20

Gleichwohl: Wenn es zu weiteren Feststellungen zur Frage der Technizität softwareimplementierbarer Erfindungen kommt, dann könnte man dabei wenigstens als ein Indiz das nehmen, was dem Sprachgebrauch der betroffenen Personen zu entnehmen ist. Im Jargon der Softwareschaffenden findet man:

25

Begriff	Treffer in google.de (Stand Oktober 2003)
Data mining	2.560.000
Virtual reality	2.160.000

Data warehous(ing)	956.000
Software tool	280.000
Software interface	88.600
Softwareschmiede	28.000

Der Sprachgebrauch zu softwarebezogenen Sujets erinnert mit „tool“, „mining“ und „Schmiede“ stark an die primären und sekundären Sektoren der industriellen Wertschöpfung: Daten werden im Jargon als Roh- und Werkstoffe gesehen, und die Werkzeuge zu ihrer Hebung und Bearbeitung werden mit Software gebaut/implementiert. Dies ist jedenfalls ein Indiz dahin, dass sich heutzutage Software als Implementierungsmittel von Erfindungen in einem weitgehend normalen, ingenieurmäßigen Kontext befindet, in dem herkömmliche technische Überlegungen wie Optimierung der Ressourcennutzung, Verbilligung, Steigerung der Zuverlässigkeit, Schaffung neuer Angebote durch Zusammenführung technischer Einzelstrukturen u. ä. angestellt werden und dann zu bestimmten Lösungen führen, die (auch) mit Software implementiert werden. Die sich daraus ergebenden ökonomischen und (patent-)rechtlichen Interessenlagen sind dann auch weitgehend „normal“, so dass aus Sicht des Autors jedenfalls nicht erkennbar ist, dass angesichts von Software als Implementierungsmittel von Erfindungen eine A-priori-Technizitäts-Skepsis zu solchen Erfindungen angebracht wäre.

Eine zeitgemäße Technikdefinition müsste nach Auffassung des Autors die Tatsache auffangen, dass Software ein „Werkstoff“ geworden ist, der in Zusammenhang steht, wie vor vielleicht 25 Jahren die Digitaltechnik sie sah, vor 45 Jahren die Röhrentechnik und davor die Relais-technik. Damit soll nicht zugunsten der Technizität von Software selbst geredet werden, denn auf die kommt es ja, da Software niemals die Erfindung ist, nicht an. Vielmehr soll gesagt werden, dass

- man sich nicht von Software den Blick auf die Technizität der von ihr implementierten Erfindungen verstellen lassen soll,
- der Umgang und das Arbeiten mit und das Reden über Software aber bestimmte Rückschlüsse auf die Technizität der implementierten Erfindungen zulässt bzw. ausschließt, wobei
- es durchaus eine nicht-triviale Aufgabe sein kann, hilfreiche Kriterien konkret zu entwickeln.

6.3 Die Kriterien „erfinderische Tätigkeit“ und „Klarheit“

Jenseits der EU-Richtlinie wäre durchaus angebracht zu untersuchen, ob und inwieweit die „Angst“ vor einem Gestrüpp von obskuren Trivialpatenten, wie sie in 3.15 angesprochen ist, gerechtfertigt ist. Patentrechtlich betrachtet wäre dies eine Untersuchung dahingehend,

- ob bei der amtlichen Sachrecherche zu Patentanmeldungen die Rechercheergebnisse hinreichend vollständig sind,
- ob angesichts des Rechercheergebnisses das Kriterium der erfinderischen Tätigkeit bei der Beurteilung von Erfindungen und der Erteilung von Patenten in sinnvoller Weise angewendet wird, und
- ob die im EPÜ explizit formulierte und im deutschen Patentgesetz implizit auch enthaltene Forderung nach Klarheit des patentierten Gegenstands hinreichend beachtet wird.

Sollten sich Unzulänglichkeiten zeigen, wären diese tatsächlich Probleme, die abgestellt werden müssen. Mit Software hat das aber nur noch wenig zu tun – wodurch sich ganz zuletzt das Fragezeichen in der Überschrift dieses Artikels rechtfertigt.