

Kurzgutachten

Wechselbeziehung zwischen Interoperabilität, Patentschutz und Wettbewerb

Dienstleistungsvorhaben Nr. 49/04

im Auftrag des Bundesministerium für Wirtschaft und Arbeit

Stand: 16. November 2004



Kontakt: Prof. Dr. Norbert Pohlmann
Fachhochschule Gelsenkirchen
Fachbereich Informatik
Neidenburgerstr. 43
45877 Gelsenkirchen

Fon: 0209 / 9596 515

Fax: 0209 / 9596 490

E-Mail norbert.pohlmann@informatik.fh-gelsenkirchen.de

Prof. Dr. iur. Andreas Möglich
Fachhochschule Gelsenkirchen
Fachbereich Wirtschaftsrecht
August Schmidt Ring 10
45665 Recklinghausen

02361 / 915 430

02361 / 915 500

andreas.mueglich@fh-gelsenkirchen.de

Inhaltsverzeichnis

KURZGUTACHTEN	I
1 EINLEITUNG.....	1
1 EINLEITUNG.....	1
2 GANG DER UNTERSUCHUNG.....	3
3 SOFTWAREMARKT	3
3.1 Marktstruktur der Softwarebranche	3
3.2 Software.....	5
4 INTEROPERABILITÄT ZWISCHEN SOFTWAREMODULEN.....	9
4.1 Interoperabilität in der IT-Wirtschaft	9
4.2 Softwaresysteme.....	10
4.3 Softwaremodule.....	10
4.4 Definition: Interoperabilität von Softwaremodulen	14
5 SOFTWARESYSTEME UND INTEROPERABILITÄT	15
5.1 Betriebssysteme.....	15
5.1.1 Ziele und Aufgaben	15
5.1.2 Diskussion über die Abhängigkeit der Interoperabilität.....	16
5.2 Kommunikationssysteme.....	19
5.2.1 Ziele und Aufgaben	19
5.2.2 Diskussion über die Abhängigkeit der Interoperabilität.....	21
5.3 Verteilte Systeme.....	22
5.3.1 Ziele und Aufgaben	22
5.3.2 Diskussion über die Abhängigkeit der Interoperabilität.....	24
5.4 Anwendungssysteme.....	26
5.4.1 Ziele und Aufgaben	26
5.4.2 Diskussion über die Abhängigkeit der Interoperabilität.....	26
5.5 Trusted Computing, Nexus.....	27
5.5.1 Ziele und Aufgaben	27
5.5.2 Diskussion über die Abhängigkeit der Interoperabilität.....	31
5.6 Kryptoalgorithmen, Sicherheitsprotokolle und Sicherheitsinfrastrukturen	32
5.6.1 Ziele und Aufgaben	33
5.6.2 Diskussion über die Abhängigkeit der Interoperabilität.....	34

5.7	Embedded Systems	35
5.7.1	Ziele und Aufgaben	35
5.7.2	Diskussion über die Abhängigkeit der Interoperabilität.....	36
5.8	Zusammenfassung - Softwaresysteme und Interoperabilität	36
6	TECHNISCHE VORAUSSETZUNGEN UND RAHMENBEDINGUNGEN DER HERSTELLUNG VON INTEROPERABILITÄT	41
6.1	Konvertierungsprogramme, Nachrichten-Broker	41
6.1.1	Beschreibung der Aufgabenstellung.....	41
6.1.2	Lösungen	42
6.1.3	Diskussion über die Herstellung von Interoperabilität.....	44
6.2	Hardwareinterfaces	45
6.2.1	Beschreibung der Aufgabenstellung.....	45
6.2.2	Diskussion über die Herstellung von Interoperabilität.....	46
6.3	Software Reverse Engineering, Dekompilation von Software	46
6.3.1	Beschreibung der Aufgabenstellung.....	46
6.3.2	Diskussion über die Herstellung von Interoperabilität.....	46
6.4	Standardisierung und Normung	47
6.4.1	Beschreibung der Aufgabenstellung.....	47
6.4.2	Diskussion über die Herstellung von Interoperabilität.....	50
6.5	Offen Schnittstellen	50
6.5.1	Beschreibung der Aufgabenstellung.....	50
6.5.2	Diskussion über die Herstellung von Interoperabilität.....	50
6.6	A rough consensus of a running code	50
6.7	Testsysteme für die Interoperabilität	51
6.7.1	Beschreibung der Aufgabenstellung.....	51
6.7.2	Diskussion über die Herstellung von Interoperabilität.....	52
7	ZUSAMMENFASSUNG: INTEROPERABILITÄT	52
8	RECHTSRAHMEN UND WIRKUNG VON SCHUTZRECHTEN IM RAHMEN DER HERSTELLUNG VON INTEROPERABILITÄT	53
8.1	Softwaresysteme und Schnittstellen	53
8.2	Schutzrechte - Patent- und Urheberrechtsschutz	54
8.2.1	Voraussetzungen des Patentschutzes.....	55
8.2.2	Patentschutz für „Computerimplementierte Erfindungen“ nach PatG	59
8.2.3	Genehmigungspraxis des DPMA	60
8.2.4	Patentschutz für „Computerimplementierte Erfindungen“ nach dem EPÜ	63
8.2.5	Zusammenfassung der Entscheidungspraxis bei „computerimplementierten Erfindungen“	67
8.2.6	Patentfähigkeit von Softwaresystemen	69
8.2.7	Zwischenergebnis: Patentschutz von Softwaresystemen	76
8.3	Wirkung ausländischer Patente - Territorialitätsgrundsatz	76

9	INTEROPERABILITÄT UND PATENTRECHT	78
9.1	Wirkung des Patentschutzes.....	78
9.2	Offenlegung - Quellcode im Patenterteilungsverfahren.....	78
9.3	Benutzung i.S.d. § 9 PatG.....	79
9.4	Interoperabilität durch Reverse Engineering.....	80
9.4.1	Begriff des Reverse Engineering	81
9.4.2	Reverse Engineering im Patentrecht.....	82
9.4.3	Reverse Engineering nach U.S.-amerikanischem Recht	88
9.4.4	Reverse Engineering im indischen Immaterialgüterrecht.....	89
9.4.5	Rechtsvergleichendes Fazit zum Reverse Engineering	90
9.5	Herstellung von Interoperabilität durch Zwangslizenzen nach § 24 PatG.....	91
9.6	Herstellung von Interoperabilität durch das Wettbewerbsrecht.....	92
9.6.1	Patentlizenzenverträge	94
9.6.2	Patentgemeinschaften und Cross-Lizenzvereinbarungen.....	95
9.6.3	Behinderung durch Sperrpatente	95
9.6.4	Schutzrechtshäufung	96
9.6.5	Anwendungsfall Microsoft	96
9.7	Herstellung von Interoperabilität durch Lizenzvertrag – Beispiel CIFS License Agreement, Fa. Microsoft.....	97
9.8	Interoperabilität von Softwaresystemen – ausgewählte Beispiele.....	99
9.8.1	Trusted Computing, Nexus	100
9.8.2	Konvertierungsprogramm/Nachrichten-Broker.....	102
9.8.3	Embedded Systems	102
9.9	Zusammenfassung – Interoperabilität und PatG	102
10	MEILENSTEINE DES RICHTLINIENENTWURFS.....	105
10.1	Grundannahmen des Richtlinienentwurfes.....	106
10.2	Übertragbarkeit urheberrechtlicher Grundsätze auf das Patentrecht?.....	107
10.3	Die Diskussion des Richtlinienentwurfes unter dem Blickwinkel „Interoperabilität“	108
10.4	Sicherstellung von Interoperabilität im Richtlinienentwurf des Kommission....	111
10.4.1	Kommissionsvorschlag.....	111
10.4.2	Änderungsvorschläge des Europäischen Parlamentes	112
10.4.3	Kompromissvorschläge	114
10.4.4	Gemeinsamer Standpunkt.....	114
11	ZUSAMMENFASSUNG UND EMPFEHLUNGEN.....	115

1 Einleitung

Die Wechselbeziehung zwischen Interoperabilität, Patentschutz und Wettbewerb ist ein sehr komplexes, vielfältigen Einflüssen ausgesetztes Thema. Aufgrund der Tatsache, dass die Diskussion um die gesetzliche Regelung der Patentfähigkeit von computerimplementierten Erfindungen nicht nur äußerst kontrovers, sondern auch in hohem Maße emotional geführt wird, soll mit dem Gutachten der Versuch unternommen werden, in der derzeit angespannten Situation, zu einer Versachlichung der Debatte beizutragen.

Nicht zuletzt das von der Europäischen Kommission gegen die Microsoft Corporation auf der Basis des EU-Wettbewerbsrechts durchgeführte Verfahren wegen wettbewerbsbeschränkender Verhaltensweisen und Missbrauch einer marktbeherrschenden Stellung hat aufgezeigt, welche überragende Bedeutung die Herstellung von Interoperabilität für den Softwaremarkt in Europa hat. Die EU-Kommission stellte sinngemäß fest, dass Interoperabilität, d.h. die Möglichkeit, über PC mit dem Server zu kommunizieren, die Grundlage für die moderne Informationstechnik darstelle. Eine Grundvoraussetzung für die Funktionsfähigkeit von Netzwerken ist in diesem Zusammenhang die Kommunikationsfähigkeit der jeweils zum Einsatz gelangenden Betriebssysteme. Hierfür ist erforderlich, dass die Schnittstellen (Interfaces) der entsprechenden Betriebssysteme angesprochen werden können. Möglich ist dies nur, wenn die entsprechenden technischen Informationen zugänglich sind¹. Gerade diese Informationen wollte Microsoft Wettbewerbern vorenthalten.

Zur Frage des Patentschutzes von computerimplementierten Erfindungen und Software sind bereits eine Reihe von Untersuchungen veröffentlicht worden. Auf die dort erzielten Ergebnisse wird, soweit für den konkreten Untersuchungsgegenstand relevant, Bezug genommen². Neuere Entwicklungen und Aspekte, die uns noch nicht ausreichend bewertet erscheinen, werden daher in dieser Untersuchung im Vordergrund stehen. Er ist somit nicht beabsichtigt, ein komplettes Abbild der Rechtsfragen bezüglich der Patentfähigkeit computerimplementierter Erfindungen insgesamt zu erstellen.

Ein wesentlicher Grund für die überaus kontrovers geführte Diskussion zwischen Befürwortern und Gegnern über die Notwendigkeit einer gesetzlichen Fixierung der Patentfähigkeit von computerimplementierten Erfindungen resultiert aus der Unsicherheit, welche Wirkungen letztlich aus der Entwicklung von Patentportfolios resultieren werden. Nicht zuletzt zahlreiche Beispiele aus den U.S.A., die Patentstreitigkeiten in diesem Sektor betreffen beunruhigen diejenigen, die sich u.U. Ansprüchen von Patentinhabern künftig ausgesetzt sehen. Selbst die nach §§ 44 ff. GWB eingerichtete Monopolkommission, deren Mitglieder der Bundespräsident auf Vorschlag der Bundesregierung beruft, äußert in ihrem Hauptgutachten 2000/2001 mit Blick auf eine mögliche Ausdehnung des Patentschutzes von Computerprogrammen erhebliche Bedenken und nimmt dabei auch auf die in Beratung befindliche EU-Richtlinie (Vorschlag für eine Richtlinie des Europäischen Parlaments und des Rates über die Patentierbarkeit computerimplementierter Erfindungen)³, Bezug⁴.

¹ Pressemitteilung der EU-Kommission vom 31. Juli 2000, IP/00/906.

² Förderung der Innovation durch Patente – Grünbuch über das Gemeinschaftspatent und das Patentschutzsystem in Europa, KOM (1997) 314 endg. Vom 24. Juni 1997; *Hart/Holmes/Reid*, The Economic Impact of Patentability of Computer Programs, *Tang/Adams/Paré*, Patent protection of computer programmes; *Blind/Edler/Nack/Straus*, Mikro- und makroökonomische Implikationen der Patentierbarkeit von Softwareinnovationen: Geistige Eigentumsrechte in der Informationsrechnologie im Spannungsfeld von Wettbewerb und Innovation, 2001; *Lutterbeck*, Sicherheit in der Informationstechnologie und Patentschutz für Software-Produkte – Ein Widerspruch?, 2000.

³ KOM (2002) 92 endg.

Zum dem wird die Diskussion um den Aspekt der Interoperabilität verkürzt, indem der Focus der Diskussion um computerimplementierte Erfindungen auf Softwareanwendungen und Sachverhalte aus dem Sektor des eCommerce eingegrenzt wird⁵

Vor diesem Hintergrund erscheint es somit als dringend erforderlich die Diskussion zu versachlichen. Diesen Anspruch verfolgt das vorliegende Gutachten, indem die Voraussetzungen für die Herstellung von Interoperabilität unter dem nationalen und europäischen Patentrechtsregime untersucht werden.

Die wichtigsten Ergebnisse der Untersuchung sind zusammenfassend folgende:

- Im Bereich der Software ist es gängige Praxis und auch eine wichtige Notwendigkeit, dass unterschiedliche Softwaresysteme wie z.B. Betriebssysteme, Anwendungen, Tools (Libraries oder sonstige Dienstroutinen, wie Middleware, usw.) und IT-Services, miteinander verwoben sind, um gemeinsam für ein Gesamtsystem sehr unterschiedliche Aufgabenstellungen zu lösen.
- Die Vitalität und Innovationsfähigkeit des deutschen Softwaremarktes hängt wesentlich davon ab, dass durch niedrige Markteintrittsschwellen eine Vielzahl von Marktteilnehmern in der Lage ist, auf der Grundlage hohen Branchen-Know-hows und entsprechender Kundennähe Individuallösungen erstellen zu können. Dies ist nur in Abhängigkeit von ausreichender Interoperabilität zu Standardbasis-Softwarelösungen (Betriebssysteme und Office-Anwendungen) gegeben.
- Berücksichtigung finden muss, dass Standardbasissoftware, wie Betriebssysteme und Office-Programme, üblicherweise von US-amerikanischen Unternehmen entwickelt und vertrieben werden, während deutsche Softwareunternehmen zum größten Teil Individuallösungen erstellen.
- Es besteht eine wachsende Notwendigkeit der Kommunikation zwischen Softwaremodulen, da absehbar ist, dass die Vernetzung von unterschiedlichen Rechnersystemen über verteilte Anwendungen notwendig sein wird.
- Die wachsende Zahl von patentierten computerimplementierten Erfindungen hat bereits dazu geführt, dass Patentinhaber vermehrt dazu neigen, produktferne Märkte durch Nichtweitergabe von interoperabilitätsrelevanten Informationen zu kontrollieren und zu monopolisieren.
- Das Patentrecht ermöglicht es Anmeldern, das Erfordernis der Offenbarung der Erfindung (§ 34 Abs. 4 PatG) durch die Vorlage des Quellcodes erfüllen zu können. Wäre dies gängige Praxis im Rahmen der Patentierung von computerimplementierten Erfindungen und würden die offen gelegten Quellcodes in der für die Herstellung von Interoperabilität erforderlichen Qualität vorliegen, wäre Interoperabilität im Regelfall gewährleistet. Die Praxis ist jedoch eine andere.
- Entwicklungsbehinderungen durch nicht Offenlegung von Informationen kann im Wesentlichen auf zwei Wegen erzielt werden. Einmal durch Reverse Engineering, das patentrechtlich nur in Fällen privater Benutzung als zulässig erachtet werden kann, und zum anderen mittels zwangsweiser Offenlegung, unter Zuhilfenahme des Kartellrechts. Urheberrechtliche Grundsätze können auf das Patentrecht wegen der bestehenden Systemunterschiede nicht

⁴ Hauptgutachten der Monoplokommission 2000/2001 – Netzettbewerb durch Regulierung, S. 349f.

⁵ Vgl. Busche, Patentschutz für computerimplementierte Erfindungen 2003, <http://www.jura.uni-duesseldorf.de/dozenten/busche/Vortragmat/softwarevortragEPA.pdf> (letzter Abruf 17. September 2004).

übertragen werden, so dass das urheberrechtliche Dekompilierungsrecht, dessen Zweckrichtung gerade die Erzeugung von Interoperabilität ist, keine Anwendung finden kann. kartellrechtliche Maßnahmen nehmen zu viel Zeit in Anspruch, wie der aktuelle Fall „Microsoft“ zeigt, vermag somit nicht die in der Informationstechnik notwendig kurzfristig Abhilfe zu schaffen.

- Besonderheiten besehen im Fall von Embedded Systems. Hier ist im Regelfall ein Revers Engineering aus technischen Gründen tatsächlich nicht möglich. Interoperabilität kann hier somit nur über die Offenlegung von Informationen erfolgen. Das Beispiel TCG zeigt, dass Standardisierungs- und Normungszwänge zur Offenlegung gerade dieser Informationen führen können.
- Der Vorschlag für eine Richtlinie des Europäischen Parlaments und des Rates über die Patentierbarkeit computerimplementierter Erfindungen enthält in seiner derzeitigen Fassung keine Ansätze zur Herstellung von Interoperabilität, die den Anforderungen der Praxis genügen würden. Interoperabilität setzt entweder die Offenlegung von Informationen, oder die Berechtigung zur Durchführung von Maßnahmen des Reverse Engineering voraus. Daher besteht entsprechender Anpassungsbedarf, dem in weiteren Teilen durch die Übernahme der Änderungsvorschläge des Europäischen Parlaments Genüge getan wäre.

2 Gang der Untersuchung

Das Kurzgutachten ist in drei Abschnitte untergliedert. Zunächst werden die informationstechnischen Grundlagen und Voraussetzungen für die Herstellung von Interoperabilität bei der Erstellung von Softwaresystemen erläutert. In einem zweiten Schritt schließt sich der Darstellung der rechtlichen Rahmenbedingungen *de lege lata*, unter Einbeziehung der Diskussionen um den Vorschlag für eine Richtlinie über die Patentierbarkeit computerimplementierter Erfindungen an, wobei im Rahmen der Bewertung der bis dato gewonnenen Erkenntnisse rechtsvergleichende Aspekte einbezogen werden. Abschließend werden in einem dritten Abschnitt mögliche Folgerungen mit Blick auf einen möglichen Anpassungsbedarf der o.g. Richtlinie erörtert.

3 Softwaremarkt

In diesem Kapitel werden einige Aspekte des Softwaremarktes dargestellt, die in Bezug auf die Diskussion über Interoperabilität von Interesse sind.

3.1 Marktstruktur der Softwarebranche

Aus der Untersuchung "Analyse und Evaluation der Softwareentwicklung in Deutschland" für das Bundesministerium für Bildung und Forschung (BMBF) können die folgenden Informationen entnommen werden⁶.

In Deutschland waren in 2000 in rund 19.200 Unternehmen der untersuchten Branchen – 10.550 Unternehmen der Primärbranchen (DV-Dienstleistern, Hersteller von Datenverarbeitungsgeräten und –einrichtungen) und 8.650 Unternehmen der Sekundärbranchen (Maschinen-

⁶ http://www.isi.fhg.de/iuk/dokumente/evasoft_abschlussbericht.pdf.

bau, Elektrotechnik, Fahrzeugbau, Telekommunikation und Finanzdienstleistungen) - Software entwickelt bzw. angepasst. In diesen Unternehmen sind insgesamt ca. 2,8 Millionen Erwerbstätige beschäftigt (ca. 300.000 Erwerbstätige in der Primärbranche und 2,5 Millionen in den Sekundärbranchen). Während die Struktur der Primärbranche überwiegend durch kleine Unternehmen mit 1-9 Mitarbeitern geprägt ist, findet Softwareentwicklung und -anpassung in den Sekundärbranchen eher in mittleren und größeren Unternehmen statt.

Der Umsatz mit Software hat sich in den letzten Jahren massiv erhöht, ein Ende des Wachstums ist nicht abzusehen. Im deutschen Markt für Software und IT-Services wurden lt. Angabe von Bitkom im Jahr 1999 allein 55,5 Mrd. DM umgesetzt⁷. In den Sekundärbranchen ist das Umsatzvolumen durch Software weit schwieriger einzuschätzen. Mittlerweile ist in vielen Sekundärbranchen der gesamte Umsatz von Software abhängig. Beispielsweise können heute in der Automobilindustrie keine Steuerfunktionen ohne Software realisiert, keine Automobile ohne Software produziert oder vertrieben, und keine betrieblichen Abläufe – von der Produktions- bis zur Personalplanung – ohne Software aufrechterhalten werden.

Als Ergebnis der repräsentativen Unternehmensbefragung wurde eine durch Softwareentwicklung bzw.-anpassung erzielte Wertschöpfung von ca. 50 Mrd. DM in den Primär- und Sekundärbranchen insgesamt errechnet. Dieser Wert ist für sich allein betrachtet bereits relativ hoch. Er übersteigt z.B. die im Sektor Land-, Forstwirtschaft und Fischerei erzielte Wertschöpfung von ca. 42 Mrd. DM um rund 20%.

Die in der Volkswirtschaft übliche und auch im vorliegenden Bericht angewendete Berechnung der Wertschöpfung wird allerdings der strategischen Bedeutung von Software - vor allem in den Sekundärbranchen - nicht voll gerecht.

Es zeigt sich an der Behandlung von Softwareentwicklungs-Abteilungen als reine Kostenstellen in der überwiegenden Zahl der Unternehmen, dass die strategische Bedeutung von Software als Umsatzgenerator auch noch nicht überall erkannt wird. Erste Unternehmen aus den Sekundärbranchen beginnen aber bereits damit, Softwareentwicklungs-Projekte nicht über ihren Kostenanteil an der Produktentwicklung, sondern über den mit Hilfe dieser Software realisierbaren Umsatz zu rechtfertigen. Damit zeichnet sich ab, dass Unternehmen Software zunehmend als Investition und „Business-Enabler“ und nicht länger als reinen Kostenfaktor begreifen.

Software entsteht als eigenständiges Produkt (Primärbranche) oder eingebettet in Produkte oder Dienstleistungen (Sekundärbranchen).

Die **Primärbranche** ist durch junge Unternehmen gekennzeichnet (67% der Unternehmen in der Primärbranche wurden nach 1990 gegründet). Neugründungen in Form von Spin-Offs entstehen durch Mitarbeiter aus Universitäten, Forschungseinrichtungen und Unternehmen, aber auch dadurch, dass etablierte Unternehmen aus ihrem internen Projektgeschäft heraus Produkte entwickeln und diese in Ausgründungen vermarkten. Damit kreieren Unternehmen traditioneller Branchen (z.B. Handel) durch Professionalisierung ihrer internen Produkte (z.B. Warenwirtschaftssysteme) neue Standbeine in der Primärbranche. Insgesamt ist davon auszugehen, dass der Boom in der Primärbranche auch auf den geringeren Investitionsbedarf dienstleistungsorientierter Unternehmen sowie auf räumliche Flexibilität zurückzuführen ist.

In den **Sekundärbranchen** spiegelt sich die traditionelle Stärke der deutschen Wirtschaft wider. Ingenieurmäßige Individuallösungen (z.B. Steuerungsfunktionen zur Sicherheit von Automobilen wie ABS, umweltschonender Energieverbrauch wie Einspritzsysteme, oder komfortable

⁷ <http://www.bitkom.org/files/documents/ITK-Marktzahlen.pdf>.

Wartung wie Online-Diagnosesysteme) tragen dazu bei, dass diese Branchen ihre Weltmarktführerschaft ausbauen können. In all diesen Branchen kann im Entwicklungsbereich eine immer stärkere Verschiebung von der Hard- zur Software beobachtet werden. Insgesamt liegt aber auch in den Sekundärbranchen das Wachstumspotenzial im Bereich der Dienstleistungen. Dies gilt nicht nur für die Unterstützung von Geschäftsprozessen in den traditionellen Dienstleistungsbranchen wie Handel (z.B. Electronic Business), Banken (z.B. Electronic Banking) und Telekommunikation, sondern manifestiert sich immer mehr durch produktbegleitende Mehrwertdienstleistungen auf der Basis von Software.

3.2 Software

Software kann unterschieden werden nach der Art der Verwendung, der Variantenvielfalt und der Installationshäufigkeit:

- Art der Verwendung

Mit Blick auf die Verwendung kann man direkt und indirekt verwendete Softwareprodukte unterscheiden. Direkt verwendete Softwareprodukte sind beispielsweise Textverarbeitungsprogramme, Tabellenkalkulationen oder Computerspiele. Sie werden in der Regel vom Nutzer installiert und sind unmittelbar als Software erkennbar. Indirekt verwendete Software wird vom Benutzer häufig nicht als Software erkannt, da sie in Produkte bzw. Dienstleistungen integriert sind. Beispiele für indirekt verwendete Software sind Steuerungen in Maschinen und Fahrzeugen, Kommunikationssoftware in Telefonen und Vermittlungen oder auch Geldautomaten in Banken. Solche Systeme sind für den Benutzer weitgehend intransparent, d.h. es ist für ihn völlig unerheblich, ob die Funktionalität durch eine Software oder auf andere Weise realisiert ist.

- Variantenvielfalt

Hierbei unterscheidet man zwischen generischen bzw. originären und individuell angepassten Softwareprodukten. Generische Softwareprodukte werden jedem Benutzer identisch zur Verfügung gestellt; eventuell sind noch benutzergeführte Einstellungen möglich. Beispiele solcher Systeme sind Textverarbeitungsprogramme oder Computerspiele. Individuell angepasste Softwareprodukte werden für jeden Kunden oder für Kundengruppen maßgeschneidert entwickelt. Dies ist beispielsweise bei kundenspezifischen Warenwirtschaftssystemen, Steuerungssystemen für Kraftfahrzeuge oder bei Kommunikationssoftware für Mobiltelefone der Fall.

- Häufigkeit der Installationen

Bei der Installationshäufigkeit kann man zwischen Systemen, die einmalig, in Kleinserien oder als Massenprodukte verkauft und installiert werden, unterscheiden. Ein Individualsystem ist z.B. die Steuerungssoftware für die Gepäckbänder an einem Flughafen. Beispiele für Kleinserienprodukte sind Finanzbuchhaltungssysteme für Banken. Massenprodukte sind u.a. Textverarbeitungssysteme, Betriebssysteme, allgemeine Geschäftsprozessunterstützungssysteme wie ERP, Handelsplattformen und Billing-Systeme oder Steuerungssoftware für einen Automobiltyp.

Besonderheit:

Es sind nur sehr wenige Softwareunternehmen im Massenproduktgeschäft (abgesehen von z.B. der SAP AG in Walldorf oder die IDS Scheer AG in Saarbrücken) tätig, das traditionell von international operierenden Unternehmen beherrscht wird. Die meisten deutschen Softwarehäuser (= softwareentwickelnde Unternehmen der Primärbranche) stellen **maßgeschneiderte Anwendungssoftware** her. Dies gilt auch für die softwareentwickelnden Unternehmen der Sekundärbranchen. Auf diesem Markt besteht die Chance, die deutsche Spitzenposition weiter zu sichern, da die Entwicklung von Anwendungssoftware in Deutschland von der Nähe zu den Anwendungsbranchen profitiert.

	In Unternehmen mit...				Gesamt
	1 – 9 Mitarbeitern	10 – 49 Mitarbeitern	50 – 199 Mitarbeitern	500+ Mitarbeitern	
Primärbranche	44.000	39.000	42.000	170.000	295.000
Sekundärbranchen gesamt	25.000	56.700	131.900	2.274.300	2.457.900
Maschinenbau	2.000	24.700	61.000	1.254.500	1.342.200
Elektrotechnik	7.300	16.100	24.500	176.000	223.900
Fahrzeugbau	150	200	3.300	136.600	142.250
Telekommunikation	250	1.500	3.400	186.000	191.150
Finanzdienstleistungen	15.300	14.200	39.700	519.200	558.400
Gesamt	69.000	95.700	173.900	2.444.300	2.782.900

Abbildung: Anzahl Mitarbeiter in Unternehmen mit Softwareentwicklung

Die Marktstruktur der Softwarebranche

Es gibt derzeit keine amtliche Datenquelle, die Informationen über Anzahl, Art (Ein- und Mehrbetriebsunternehmen), Größe und Branchenzugehörigkeit der Unternehmen und Arbeitsstätten in Deutschland vollständig und aktuell abdecken. Von der amtlichen Statistik ist diese Grundgesamtheit zuletzt 1987 in der Arbeitsstättenzählung erhoben worden. Für die meisten Branchenstudien haben diese Zahlen heute nur noch wirtschaftshistorischen Wert. Insbesondere trifft dies auf den stark wachsenden Dienstleistungssektor zu. Des Weiteren ist eine komplette Erfassung der Arbeitsstätten in den östlichen Bundesländern überhaupt nicht vorhanden.

Die hier dargestellte Branchenstudie versucht daher, auf Grundlage einer Primärerhebung und den in Experteninterviews ermittelten Aussagen diese Lücke zu füllen und ein aktuelles Bild der bedeutendsten softwareentwickelnden Branchen in Deutschland zu liefern.

Unternehmen und Branchen

Die in dieser Studie ermittelten Daten bilden die Grundlage für die Darstellung des Marktes der Softwareentwicklung in Deutschland. Dabei wird differenziert nach den Grundgesamtheiten der Primärbranche und den Sekundärbranchen; nachfolgend werden diese Teilsegmente vertiefend beschrieben.

Unternehmen der Primärbranche

Folgende Ausgangsbasis wurde für die Hochrechnung der Unternehmen in der Software-Primärbranche zugrunde gelegt (vgl. Kapitel 2.3.2):

mit 1 bis 9 Beschäftigten	32.693 Unternehmen
mit 10 bis 49 Beschäftigten	2.444 Unternehmen
mit 50 bis 199 Beschäftigten	475 Unternehmen
mit 200 und mehr Beschäftigten	185 Unternehmen
Gesamt	35.797 Unternehmen

Unternehmen der Sekundärbranchen

Der große Stellenwert des produzierenden Gewerbes in Deutschland zeigt sich nicht zuletzt daran, dass das Statistische Bundesamt für diese Branchen zumindest für Unternehmen mit mehr als 20 Mitarbeitern aktuelle Zahlen für Unternehmen, Beschäftigten, Umsätze etc. vorhält [Statistisches Bundesamt 1998]. Grundlage für die Abschätzung der Grundgesamtheit für kleinere Unternehmen waren die Zahlen der Beschäftigtenstatistik der Bundesanstalt für Arbeit.

Daraus ergeben sich für die ausgewählten fünf produzierenden Sekundärbranchen folgende Grundgesamtheiten:

Maschinenbau

mit 1 bis 9 Beschäftigten	11.960 Unternehmen
mit 10 bis 49 Beschäftigten	7.034 Unternehmen
mit 50 bis 199 Beschäftigten	2.409 Unternehmen
mit 200 und mehr Beschäftigten	1.027 Unternehmen
Gesamt	22.340 Unternehmen

Elektrotechnik

mit 1 bis 9 Beschäftigten	16.670 Unternehmen
mit 10 bis 49 Beschäftigten	6.019 Unternehmen
mit 50 bis 199 Beschäftigten	1.632 Unternehmen
mit 200 und mehr Beschäftigten	732 Unternehmen
Gesamt	25.053 Unternehmen

Fahrzeugbau

mit 1 bis 9 Beschäftigten	1.369 Unternehmen
mit 10 bis 49 Beschäftigten	679 Unternehmen
mit 50 bis 199 Beschäftigten	321 Unternehmen
mit 200 und mehr Beschäftigten	256 Unternehmen

Gesamt 2.635 Unternehmen

Branchen im Dienstleistungsbereich werden vom Statistischen Bundesamt außerhalb der Umsatzsteuerstatistik (noch) nicht erfasst. Für die Ermittlung der Grundgesamtheiten der beiden Sekundärbranchen Telekommunikation und Finanzdienstleistungen wurden deshalb ausschließlich die Daten der Beschäftigtenstatistik der Bundesanstalt für Arbeit herangezogen. Da diese Daten allerdings wie bereits beschrieben lediglich auf der Basis von Betrieben, aber nicht auf der Basis von Unternehmen dargestellt werden, wurden hier zusätzlich Ergebnisse aus GfK-eigenen Untersuchungen herangezogen. Damit ergeben sich folgende Grundgesamtheiten:

Telekommunikation

mit 1 bis 9 Beschäftigten	731 Unternehmen
mit 10 bis 49 Beschäftigten	326 Unternehmen
mit 50 bis 199 Beschäftigten	128 Unternehmen
mit 200 und mehr Beschäftigten	115 Unternehmen

Gesamt 1.300 Unternehmen

Finanzdienstleistungen

mit 1 bis 9 Beschäftigten	91.711 Unternehmen
mit 10 bis 49 Beschäftigten	2.474 Unternehmen
mit 50 bis 199 Beschäftigten	1.401 Unternehmen
mit 200 und mehr Beschäftigten	1.943 Unternehmen

Gesamt 96.529 Unternehmen

Originäre Entwicklung und Anpassung von Software

Bei der Art der Softwareentwicklung überwiegt in der Gesamtbetrachtung derzeit die **individuelle Anpassung von Software** gegenüber der originären Entwicklung: 14.700 Unternehmen kaufen Basis-Software bei externen Anbietern und entwickeln diese intern weiter. 11.700 Unternehmen entwickeln Software originär.

Die Unterschiede zwischen Primärbranche und den fünf untersuchten Sekundärbranchen werden bei der Art der Softwareentwicklung besonders deutlich. Unternehmen der Primärbranche sind überwiegend (73%) mit der Neu- und Weiterentwicklung von Software beschäftigt. In der Sekundärbranche liegt der Schwerpunkt mit 87% der Unternehmen eindeutig auf der Individualisierung und Weiterentwicklung extern bezogener Basis-Software. Ausnahmen sind hier die Branchen Elektrotechnik (63% originäre Softwareentwicklung) und Telekommunikation (59%). Der Grund ist in der dort sehr häufig festgestellten Einschätzung zu finden, dass die Software in besonders hohem Masse zum Alleinstellungsmerkmal des Unternehmens bzw. des Produktes beiträgt. Zukünftige Leistungsmerkmale wollen die Unternehmen selbst entwickeln. Der Know-how-Schutz gegenüber den Wettbewerbern ist besonders relevant.

4 Interoperabilität zwischen Softwaremodulen

Eine besondere Frage im Bereich der Software ist, ob die Interoperabilität eine besondere Rolle für die Funktionstüchtigkeit des Wettbewerbs spielt. Im Zusammenhang mit der Auswertung der informationstechnologischen und rechtlichen Aspekte untersucht werden, ob und wie die gewonnen Erkenntnisse im Gesetzgebungsverfahren aus der Sicht des Patenrechts zu berücksichtigen sind.

Aus diesem Grund wird in diesem Abschnitt diskutiert, wie sich die Interoperabilität zwischen Softwaremodulen auf die IT-Wirtschaft auswirkt und wie die Abhängigkeiten von Softwaremodulen untereinander sind und wie Interoperabilität gewährleistet werden kann. In diesem Teil des Gutachtens wird nicht berücksichtigt, ob die Schnittstellentechnologie als computerimplementierte Erfindung verstanden werden kann oder nicht.

4.1 Interoperabilität in der IT-Wirtschaft

Im Bereich der Software ist es gängige Praxis und auch eine wichtige Notwendigkeit, dass unterschiedliche Softwaresysteme wie z.B. Betriebssysteme, Anwendungen, Tools (Libraries oder sonstige Dienststroutinen, wie Middleware, usw.) und IT-Services, miteinander verwoben sind, um gemeinsam für ein Gesamtsystem sehr unterschiedliche Aufgabenstellungen und Branchenwissen zu lösen.

Daraus ergibt sich die Notwendigkeit, über Softwareschnittstellen Informationen auszutauschen zu können.

In der Praxis können Softwaremodule in einem Rechnersystem oder über Netzwerke (z.B. mit der Hilfe von RMI, RPC über Kommunikationsprotokolle) miteinander kommunizieren. Die Notwendigkeit über Netzwerke zu kommunizieren wird durch die Umsetzung moderner IT-Konzepte in der Zukunft noch viel umfänglicher und flexibler notwendig sein. Verteilte Systeme, Mobilanwendungen und Ubiquitous- oder Pervasive-Computing sind nur einige IT-technische Stichworte, die diese Notwendigkeit aufzeigen.

Im Bereich der „Verteilten Systeme“, die wegen der Leistungsfähigkeit und Redundanz immer mehr in der Informationsgesellschaft in den Vordergrund treten, spielen Konvertierungssysteme eine besondere Rolle, weil hier unterschiedliche IT-Architekturen (z.B. Rechnersysteme wie SmartPhones, PDAs, Notebooks/PCs, Server und Großrechner sowie unterschiedliche Kommunikationsarchitekturen wie (TCP/IP, OSI, IBM-SNA, DEC-Net, ...) miteinander verbunden werden sollen.

Aus diesem Grund ist die informationstechnische Notwendigkeit der Interoperabilität von IT-Systemen aus heutiger Sicht unabdingbar und hat eine sehr hohe Bedeutung für die IT-Wirtschaft, da die Interoperabilität Unternehmen es ermöglicht, die Software aus sehr unterschiedlichen Branchen, zu einem Gesamtsystem zusammen zuführen.

4.2 Softwaresysteme

Softwaresysteme bestehen i.d.R. aus vielen unterschiedlichen Softwaremodulen, die gemeinsam eine Aufgabe erfüllen. Softwaresysteme sind z.B. Betriebssysteme, Datenbanksysteme, Verteilte Systeme, spezielle Anwendungen, usw. .

Wenn Softwaresysteme zur Verfügung gestellt werden, bedeutet das in der Praxis, dass nur der notwendige Teil der vorhandenen Schnittstellen bekannt gegeben werden muss, um die gewünschte Funktionalität des Softwaresystems nutzen zu können.

Die Komplexität der Softwareschnittstelle (Schnittstellen, Interface) ist bei den Softwaresystemen sehr unterschiedlich. Z.B. bieten Betriebssysteme komplexere Schnittstellen als spezielle Anwendungssysteme.

4.3 Softwaremodule

Jedes Softwaresystem wird durch eine Verknüpfung von mehreren Softwaremodulen zusammengestellt. In einer Informationsgesellschaft ist die Komplexität der Softwaresysteme teilweise so groß, dass es nur im Zusammenspiel mit anderen Softwaremodulen realisiert werden kann, die von anderen Firmen mit Spezial-KnowHow entwickelt werden (Betriebssysteme, Datenbanken, branchenspezifische Anwendungen, usw.). Ohne die Interoperabilität dieser Softwaremodule ist es nicht möglich, ein verwertbares Softwaresystem zur Verfügung zu stellen, d.h. dass heute jede funktionierende Software aus miteinander interoperablen Softwaremodulen besteht.

Softwaremodule sind Computerprogramme, die auf einem Rechnersystem (SmartPhones, PDAs, Notebooks/PCs, Server, Großrechner, Handys, Autos, Roboter, Industriesteuerungen, usw.) ablaufen und über eine anzusprechende Schnittstelle definierte Funktionen erbringen. Die Operationen/Funktionen können rein softwaremäßig mit Hilfe einer CPU und weiteren Komponenten eines Rechnersystems realisiert werden. Es kann aber auch sein, dass die Operationen/Funktionen mit Hilfe der Software z.B. durch eine NC-Maschine, Bremsen im Auto, usw. erbracht werden.

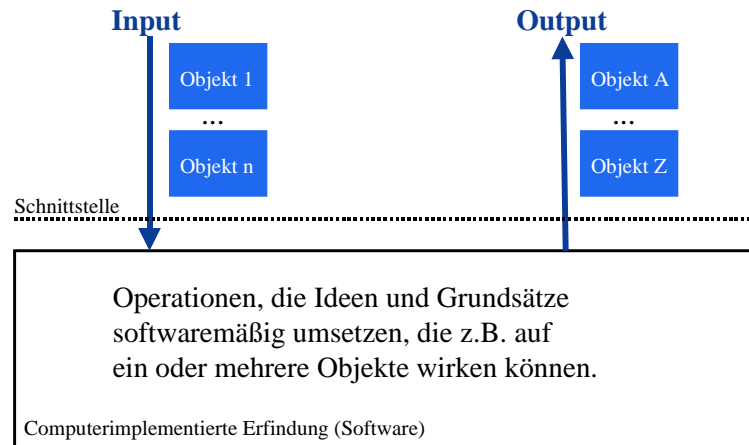


Abbildung: Softwaremodul

Softwareschnittstelle:

Über die Softwareschnittstelle oder Programmierschnittstellen kann ein anderes Softwaremodul (Datenbankprozess, Web-Server, Steuerung einer Hardware, usw.) auf die Operationen eines Softwaremoduls zugreifen und sie entsprechend nutzen.

Um diese Operationen nutzen zu können, muss die Softwareschnittstelle gut dokumentiert sein.

Wenn es keine Möglichkeit gibt, die Softwareschnittstelle zu nutzen, dann können die Operationen auch nicht genutzt werden.

Der allgemeine Nutzen von Softwareschnittstellen liegt in der Codewiederverwendung über Firmengrenzen hinweg. Dies bedeutet, dass neben der Syntax auch Semantik, Laufzeitverhalten und Fehlerverhalten etc. hinreichend präzise definiert sein müssen. Die Softwaremodule sollen so konstruiert werden, dass sie potentiell auch als binäre Produkte (d.h. ohne Quellcode) auf den Markt gebracht und vom Kunden bei minimalen Kosten integriert werden können.

D.h. auch, dass i.d.R. von Firmen nur die Softwareschnittstellen bekannt gegeben werden, die von anderen genutzt werden sollen, um die gewünschte oder beabsichtigte Funktionalität nutzen zu können.

Softwareschnittstellen werden oft auch als Application Programming Interface (API) bezeichnet. Application Programming Interface (API) sind:

- Programmierschnittstelle zum Schreiben von Anwendungssoftware auf Basis schon zur Verfügung gestellter Funktionalitäten. Solche Schnittstellen werden z.B. für Betriebssysteme und Kommunikationsprotokolle (allg. auch andere Software) bereitgestellt⁸.

oder

⁸ <http://www.lanline.de/html/lanline/lexikon/lex/api.htm>.

- Die Schnittstelle eines Programms, über die externe Applikationen dieses steuern oder auf die Inhalte zugreifen kann⁹.

oder

- Das Application Programming Interface ist eine dokumentierte Software-Schnittstelle, mit deren Hilfe ein Programm die Funktionen eines anderen Programms nutzen kann.¹⁰

Da die Software in sehr unterschiedliche Umgebungssysteme eingebunden wird und genutzt werden kann, gibt es keine einheitlichen Methoden der Beschreibung von Softwareschnittstellen. Eine sehr hilfreiche Methode, eine gute Beschreibung der Schnittstelle einer Software zur Verfügung zu stellen, ist ein "Software Development Kit (SDK)". Ein Software Development Kit, kurz SDK, ist eine Sammlung von Programmen und Dokumentation zu einer bestimmten Software, die es Software-Entwicklern erleichtern soll, eigene darauf basierende Anwendungen zu erstellen¹¹.

Firmen, die ein besonderes Interesse haben, dass andere ihre Software mit in ihre Lösung einbinden, können durch die zur Verfügungstellung eines SDK diesen Prozess erheblich vereinfachen. Durch eine sehr gute Dokumentation der Softwareschnittstelle mit Beispielen, wie eine Einbindung realisiert werden kann, können die einbindenden Firmen sehr viel Zeit und Geld sparen. Bei vielen Softwareprodukten werden SDKs auch zum Kauf angeboten, damit können die Firmen ihren Aufwand für die Erstellung eines SDK finanzieren.

In der folgenden Tabelle werden verschiedene Kategorien von Softwareschnittstellen dargestellt sowie deren Bewertung und die Anforderungen skizziert, die Voraussetzung für die Herstellung von Interoperabilität ist.

Kategorien von Softwareschnittstellen	Bewertung bezüglich der Interoperabilität	Anforderungen bezüglich der Interoperabilität
Softwareschnittstellen und Protokolle sind Standards von Normungsgremien (DIN, ISO, ITU, IETF, W3C, usw.)	Die Softwareschnittstellen und Protokolle können i.d.R. frei verwendet werden und haben im Prozess der Standardisierung einen Konsens erlangt. (siehe Kap. Normen und Standards)	Die Schnittstellen werden aus der Norm entnommen Es gibt in der Praxis unterschiedliche Konzepte, wie die Interoperabilität sichergestellt wird <ul style="list-style-type: none"> • Testsysteme (z.B. ISIS-MTT) • mehrere Lösungen von unterschiedlichen Firmen müssen nachweislich miteinander laufen, damit sie zum Standard werden (z.B. IETF)

⁹ http://www.contentmanager.de/ressourcen/glossar_6_api.html.

¹⁰ <http://www.commando.de/glossar/gloss-a.htm>.

¹¹ <http://de.wikipedia.org/wiki/SDK>.

Software mit offenen Schnittstellen	Die Schnittstelle der Softwaremodule kann genutzt werden, auch wenn sie normalerweise nicht direkt angesprochen wird.	Wichtig ist, dass die Qualität der Schnittstellenbeschreibung ausreichend gut ist (z.B. SDK).
Software ohne offene Schnittstellen	Die Schnittstelle der Softwaremodule kann nicht genutzt werden, weil sie nicht bekannt ist.	Hier müssen Reverse-Engineering-Methoden verwendet werden, um die Schnittstelle und das Verhalten zu ermitteln (siehe Kap. Software Reverse Engineering, Dekompilation von Software)

Im Folgenden wird beschrieben, in welchen Formen eine Software sich in der Praxis repräsentiert.

Quell- und Objektcode:

Als "Quellcode" bezeichnet man zum einen Programm-Anweisungen in einem Computerprogramm (engl.: "Source Code" oder "Source Program"), zum anderen aber auch den Quelltext eines Web-Dokuments (engl.: "Document Source"). Der Quellcode eines Dokuments im "World Wide Web" (WWW) besteht aus dem Text des Dokuments (Content) und den Markierungen (Markup, Tag), die ihn strukturieren und formatieren. Sowohl der im Browser angezeigte Text des Dokuments als auch die nur im Quelltext sichtbaren Tags der "HyperText Markup Language" (HTML) bestehen ausschließlich aus ASCII-Textzeichen.

Der **Quellcode eines Computerprogramms** besteht aus Anweisungen, die in einer höheren Programmiersprache oder in Assembler geschrieben sind. Sie können von Menschen gelesen oder verändert, aber nicht direkt von einem Computer verarbeitet (ausgeführt) werden. Der Quellcode eines Programms ist eine editierbare Datei, die zum Beispiel aus Textzeichen im "American Standard Code for Information Interchange" (ASCII) bestehen kann. Um aus einem Quellcode einen ausführbaren Code zu machen, muss der Quellcode von einem Compiler-, Assembler- oder Interpreter-Programm in Objektcode (engl.: "Object Code") umgewandelt (kompiliert) werden. Der Quellcode kann auch verwendet werden, um die Schnittstelle einfach zu ermitteln und damit eine Interoperabilität zu Softwareschnittstellen zu erhalten. Typischerweise erhält der Quellcode auch Kommentare, die helfen die Funktionsweise besser zu verstehen.

Die Bezeichnung "**Objektcode**" wird sowohl für Maschinencode verwendet, den der Prozessor des Computers (CPU) direkt ausführen kann (Binärcode), als auch gelegentlich in Bezug auf einen Assembler-Quellcode oder eine bestimmte Spielart eines Maschinencodes (Byte-Code). Bei Skriptsprachen wie Perl oder JavaScript sollte man streng genommen nicht von Quellcode sprechen, da der Code in diese Skriptsprachen nicht zu einem Objektcode kompiliert wird¹². Der Objektcode kann i.d.R. nicht direkt verwendet werden, um die Schnittstelle einfach zu ermitteln und damit eine Interoperabilität zur Softwareschnittstelle zu erhalten. Aus dem Quellcode lässt sich durch Interpretation oder Kompilierung einfach der Objektcode erstellen, dagegen ist es technisch aufwendig, aus dem Objektcode den Quellcode zu erlangen (siehe Kap. Software Reverse Engineering, Dekompilation von Software).

¹² siehe <http://www.lexitron.de/main.php?detail=true&eintrag=986>.

4.4 Definition: Interoperabilität von Softwaremodulen

Unter Interoperabilität von Softwaremodulen, wird die Fähigkeit von Softwaremodulen verstanden, Informationen direkt im Rechnersystem oder mit Hilfe von Protokollen über Netzwerke untereinander verarbeiten zu können, d.h. dass Softwaremodule unterschiedlicher Hersteller untereinander Informationen (Input und Output) austauschen können. Dies setzt voraus, dass Syntax und Semantik der Informationen, die zwischen den Softwaremodulen ausgetauscht werden, in einheitlicher Form zur Verfügung stehen.

Protokolle sind Vorschriften und Regeln zum Informationsaustausch zwischen zwei oder mehreren Partnern (in Rechnersystemen) auf derselben Stufe der Funktionsschichtung (ISO- oder TCP/IP-Referenzmodell) eines Kommunikationssystems. Sie bestehen aus präzisen Spezifikationen in syntaktischer, prozeduraler und semantischer Hinsicht. Dies erlaubt eine Aufteilung des Softwaremarktes unter Firmen aus verschiedenen Branchen, sowie die Möglichkeit, ein Endprodukt oder eine Gesamtlösung anhand mehrerer, spezialisierter, von verschiedenen Herstellern stammenden Softwaremodulen, herzustellen.

Informationen und deren Datenformate (z.B. herstellerspezifisch, XML, EDIFACT, u.a.), die über die Softwareschnittstelle ausgetauscht werden und Protokolle (z.B. SMTP, FTP, HTTP, usw.) müssen daher offen, d.h. deren Definition für alle zugänglich sein, damit ein Austausch realisiert werden kann¹³.

Eine weitere Definition des Begriffs der Interoperabilität ergibt sich aus dem "Beschluss 2004/387/EG DES EUROPÄISCHEN PARLAMENTS UND DES RATES vom 21. April 2004 über die interoperable Erbringung europaweiter elektronischer Behördendienste (eGovernment-Dienste) für öffentliche Verwaltung, Unternehmen und Bürger (IDABC)¹⁴:

"Interoperabilität ist die Fähigkeit von IKT-Systemen (Systeme der Informations- und Kommunikationstechnologie) und der von ihnen unterstützten Geschäftsanwendungen, Daten auszutauschen und die gemeinsame Nutzung von Informationen und Kenntnissen zu ermöglichen."

Aus den vorgenannten Ansätzen der Definition des Begriffs Interoperabilität ergeben sich im Schwerpunkt die nachfolgend genannten Fallvarianten, in denen die Interoperabilität von Softwaremodulen hergestellt wird:

- direkt im Rechnersystem wird Interoperabilität über eine Softwareschnittstelle erzielt;
- auf verschiedenen Rechnersystemen wird Interoperabilität mit Hilfe von Kommunikationsprotokollen und den auszutauschenden Objekten erreicht;
- Interoperabilität mittels gemeinsamer Speicherbereiche (File-Server, USB-Token, CD, u.a.) wird über Dateiformate erreicht.

Hieraus folgt, dass Interoperabilität in ganz unterschiedlichen Erscheinungsformen, d.h. technischen Umsetzungen, relevant wird und unter verschiedenen technischen Randbedingungen betrachtet werden muss.

¹³ siehe auch <http://www.computerlexikon.com/begriff.php?id=796> und <http://www.geoinformatik.uni-rostock.de/einzel.asp?ID=990>.

¹⁴ http://www.governet.de/themen/egov_strategien/20210.html.

5 Softwaresysteme und Interoperabilität

In diesem Abschnitt werden exemplarisch einige wichtige Softwaresysteme beschrieben und deren Abhängigkeit bezüglich der Interoperabilität diskutiert.

Im folgenden Bild wird ein einfaches Beispiel exemplarisch als Übersicht von möglichen Zusammenhängen der unterschiedlichen Softwaresysteme in einem Rechnersystem dargestellt.

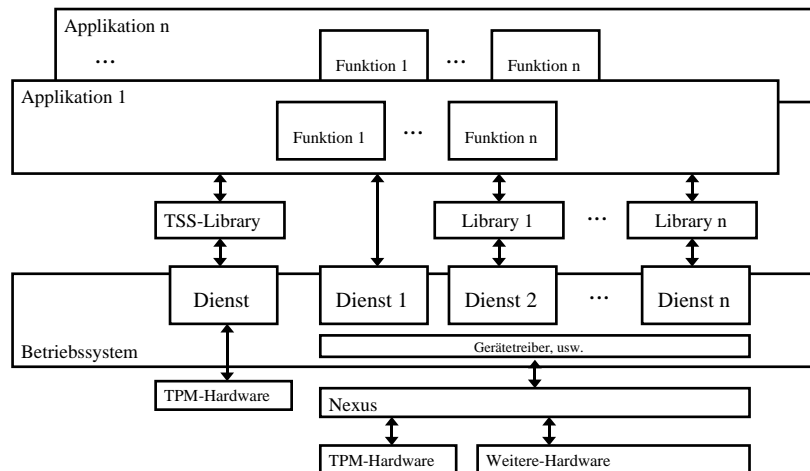


Abbildung: Zusammenhänge unterschiedlicher Softwaresysteme

Hinweis:

In diesem Bild sind 1 bis n Applikationen mit verschiedenen Funktionen aufgezeigt, die direkt oder über Libraries auf die Dienste eines Betriebssystems zugreifen, um ihre speziellen Aufgaben erfüllen zu können. Im unteren Bild ist ein neues Sicherheitssystem (Nexus) und ein Hardware sicherheitsmodul (TPM) eingezeichnet, die zukünftig in Rechnersystemen angeboten werden sollen (Siehe Kapitel "Trusted Computing, Nexus").

5.1 Betriebssysteme

Im folgenden Abschnitt wird das Softwaresystem "Betriebssystem" beschrieben und die Abhängigkeit der Interoperabilität diskutiert.

5.1.1 Ziele und Aufgaben

Ein Betriebssystem ist allgemein ein Computerprogramm, das die Ausführung von Anwendungsprogrammen steuert und als Schnittstelle zwischen den Anwendungen und der Rechner-systemhardware dient.

Es hat drei Ziele:

- **Bequemlichkeit:** Durch ein Betriebssystem kann ein Rechnersystem bequem benutzt werden.

- Effizienz: Durch ein Betriebssystem können die Rechnersystemressourcen effizient genutzt werden.
- Fähigkeit zur Weiterentwicklung: Ein Betriebssystem sollte so aufgebaut sein, dass es die effektive Entwicklung, das Testen und die Einführung neuer Systemfunktionen ermöglicht, ohne dass dies den Betrieb negativ beeinflusst¹⁵.

Im folgenden Bild wird als ein Beispiel die Windos-2000-Architektur dargestellt.

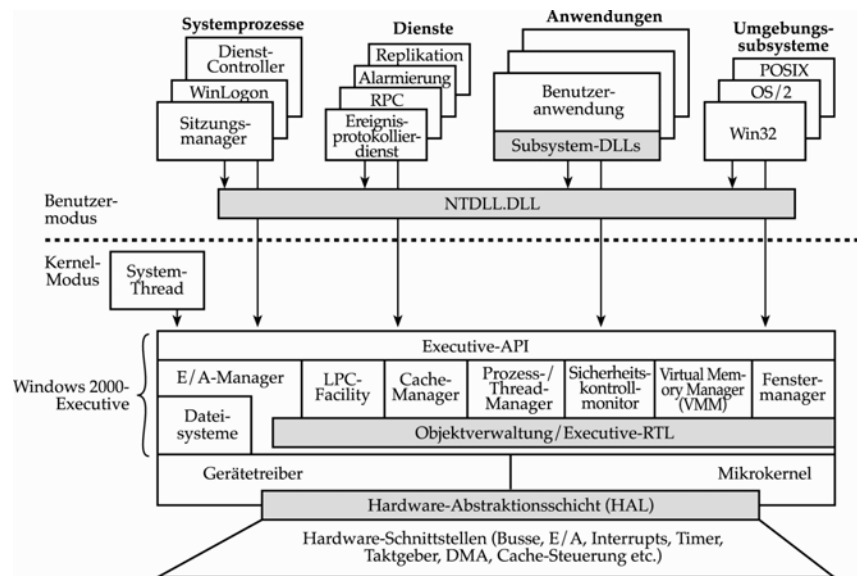


Abbildung: Windos-2000-Architektur

Hinweis:

Der Betriebssystemaufbau und die einzelnen Komponenten eines Betriebssystems können in Stallings 2004 auf Seiten 111 und folgende Seiten nachgelesen werden.

5.1.2 Diskussion über die Abhängigkeit der Interoperabilität

Jedes Softwaremodul muss letztendlich direkt oder indirekt über das Betriebssystem die Hardwarefunktionen eines Rechnersystems nutzen, um die eigentlichen Funktionen eines Softwaremoduls ausführen zu können. Aus diesem Grund ist die Abhängigkeit der Interoperabilität vom Betriebssystem und seinen Diensten in der Gesamtheit 100%. Ohne Betriebssystem kann i.d.R. kein Softwaremodul zur Ausführung gebracht werden.

Wie in dem Beispiel des Betriebssystemaufbaus vom Windows-2000 Betriebssystem zu sehen ist, sind Betriebssysteme komplexe Softwaresysteme, die mit vielen Softwaremodulen Informationen austauschen müssen, damit die Ziele und Aufgaben eines Betriebssystems insgesamt realisiert werden können. Im Fall von Windows-2000 oder anderen Microsoft Betriebssystemen wird diese von einem Hersteller nämlich Microsoft entwickelt und für den Kauf in Form von Objektcode zur Verfügung gestellt. Im Fall von dem OpenSource-Betriebssystem Linux gibt es, neben dem Kaufpreis, zwei große Unterschiede. Erstens wird das Betriebssystem von mehre-

¹⁵ Stallings, S. 74.

ren Personengruppen entwickelt und zweitens wird die Software mit allen Beschreibungen als Quellcode für alle zur Verfügung gestellt.

Der Unterschied in der Praxis ist der, dass von den Funktionen/Diensten, die das Betriebssystem von Microsoft mit Hilfe von Softwaremodulen realisiert hat, nur diejenigen einfach genutzt werden können, von denen eine gute Dokumentation zur Verfügung gestellt wird.

Marktanteile¹⁶:

Wie aus einer aktuellen Studie des Marktforschungsunternehmens Gartner hervorgeht, sind 2003 weltweit rund 155 Millionen PCs und damit 96,4 Prozent aller verkauften Geräte mit einem Windows-System ausgeliefert worden.

Das konkurrierende Open-Source-Betriebssystem Linux hat im Vorjahr laut der Studie nur geringe Zuwächse erzielen können, der Marktanteil stieg von 1,3 auf 1,5 Prozent.

Patente

Nach einer Analyse des Linux Quellcodes, die von der Open Source Risk Management Inc. veranlasst wurde, sollen im Linux Betriebssystem 283 Patentverletzungen enthalten sein¹⁷.

Hinweis:

Es ist zu bedenken, dass Microsoft selbst ganz genau weiß, welche Schnittstellen für den Markt bzw. für gewisse Branchen interessant sind, und gezielt spezielle Schnittstellen undokumentiert gelassen werden, damit dieser Markt, Branche von keiner andere Firma besetzt werden kann.

¹⁶ <http://futurezone.orf.at/futurezone.orf?read=detail&id=211494>.

¹⁷ <http://www.spiegel.de/netzwelt/netzkultur/0,1518,311542,00.html>.

Abhängigkeit von Softwareschnittstellen

Einige Firmen sind z.Z. davon abhängig, dass sie über Reverse Engineering Methoden die Softwareschnittstellen und das Verhalten der Softwaremodule analysieren, um nicht oder nicht gut dokumentierte Funktionen/Dienste im Betriebssystem nutzen zu können. In folgenden soll exemplarisch mit der Hilfe eines Beispiels dargestellt werden, warum auch auf nicht oder nicht gut dokumentierte Schnittstellen zugegriffen werden muss, um eine vom Markt gewünschte Funktionalität realisieren zu können.

Einbindung von Sicherheitssoftware für die Verschlüsselung der Daten auf der Festplatte

Die als Beispiel ausgewählte Sicherheitssoftware hat den Zweck, Festplatten zu verschlüsseln. Im Gegensatz zu Lösungen, die "nur" Dateien und Verzeichnisse verschlüsseln, kann eine Festplattenverschlüsselung die gesamte Festplatte verschlüsseln – inklusive sämtlicher Betriebssystemdateien, temporären Dateien, Hibernation-Datei, etc. Die direkte Konsequenz dieser Anforderung ist, dass der Logon für eine solche Festplattenverschlüsselung zu einem sehr frühen Zeitpunkt während des Bootens des Rechners erfolgen muss. In der Praxis geschieht das zumeist noch bevor die Bootpartition der Festplatte gelesen wird und somit entweder im Master Boot Record (MBR) oder im Bootsektor der aktiven Partition (meistens Partition C:). Ein weiteres Problem, das sich bei der Festplattenverschlüsselung stellt, ist das, wo der Code abgelegt wird, aus dem der Logon-Bildschirm besteht. Zu einem solch frühen Zeitpunkt im Bootprozess eines PCs gibt es noch kein aktives Dateisystem, welches genutzt werden kann. Es liegt also nahe, den Festplattenbereich, welcher den Logon-Bildschirm für die Eingabe einer Identität (ID) und eines Passwortes (PW) beinhaltet, auf andere Art und Weise zu belegen und zu sichern, damit die Festplattenverschlüsselung realisiert werden kann.

Eine nahe liegende Möglichkeit, das zu erreichen, ist für den Logondialog im Windows Dateisystem NTFS oder FAT Festplattenbereich zu allokierten, so dass er belegt und nicht verschiebbar ist. Diese beiden Bedingungen sind wichtig und können erfüllt werden, indem dieser Bereich als ‚Bad Cluster‘ im Dateisystem NTFS oder FAT angelegt wird. Für die Betriebssysteme Windows NT4, 2000 und XP rät Microsoft NTFS als Standard Filesystem zu verwenden.

Die Informationen vom Dateisystem NTFS, z.B. wie ‚Bad Cluster‘ aussehen oder anzulegen sind, sind nicht bzw. nur sehr grob beschrieben. Dennoch müssen Firmen, welche Festplattenverschlüsselung entwickelt wollen, in der Lage sein, die notwendigen Schnittstellen mit allen Informationen durch Reverse Engineering (Dekompilierung, Ausprobieren, usw.) herauszubekommen.

Im folgenden Bild sind die Softwareschnittstellen (dicker Balken) angedeutet, die aus Interoperabilitätsgründen analysiert werden müssen, um die gewünschte Funktionalität realisieren zu können.

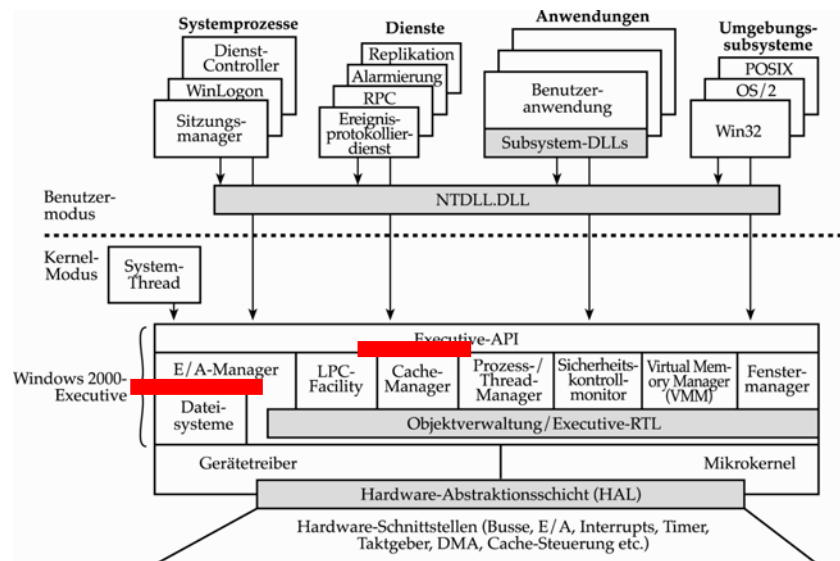


Abbildung: Schnittstellen, bei denen ein Reverse-Engineering durchgeführt werden muss.

Aufwand, der notwendig ist, um ein Festplattenverschlüsselungssystem Reverse Engineering zu betreiben

Die Vorgehensweise beim Reverse Engineering ist die, dass mit Hilfe eines (Kernel-)Debuggers nachvollzogen wird, wie andere Software auf die notwendigen Schnittstellen zugreift. Mit diesen Erkenntnissen wird die notwendige eigene Software programmiert und per "Trial and Error" so lange versucht, bis die notwendigen Funktionen ohne Fehler eingebunden werden können.

Der Aufwand, um Reverse Engineering durchzuführen, beträgt 1 bis 3 Mann-Monate, in Abhängigkeit der Erfahrungen der Personen die es durchführt und der Komplexität der Schnittstellen, die analysiert werden müssen. Ein weiteres Problem, was hier auftritt ist, dass bei jedem Release-Wechsel der Software des Betriebssystems möglicherweise die Schnittstelle verändert wurde und wieder durch Reverse Engineering die Änderungen gesucht werden müssen, damit die eigene Software entsprechend angepasst werden kann. Der Gesamtaufwand muss für jedes unterschiedliche Betriebssystem (NT, 2000, XP, usw.) durchgeführt werden.

Mit der Hilfe von Reverse Engineering kann die Softwareschnittstelle generell nie zu 100% ermittelt werden. Die Erfahrung hat gezeigt, dass es i.d.R. dennoch möglich ist, hierdurch für die Praxis eine ausreichende Interoperabilität gewährleisten zu können.

5.2 Kommunikationssysteme

Im folgenden Abschnitt wird das Softwaresystem "Kommunikation" beschrieben und die Abhängigkeit der Interoperabilität diskutiert.

5.2.1 Ziele und Aufgaben

Das Kommunikationssystem eines Rechners ist für den rechnerübergreifenden Austausch von Nachrichten verantwortlich und benutzt zur Übertragung ein **Rechnernetz**. Wenn viele Rechner

untereinander verbunden sind und Daten austauschen können, nennt man dies ein **Rechnernetz**. Ein **Rechnernetz** ist primär ein Transport- und Übertragungssystem zwischen an ein Netz angeschlossenen, weitgehend oder vollständig autonomen Rechnersystemen¹⁸.

Im folgenden Bild wird als Beispiel ein einfaches Rechnernetz dargestellt.

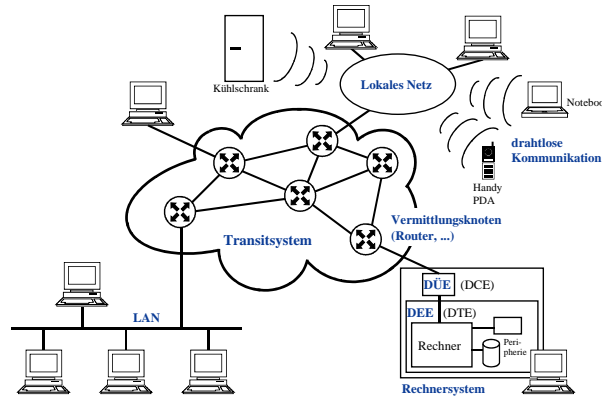


Abbildung : Beispiel eines einfachen Rechnernetzes

Ein **Rechnernetz** bietet seinen Teilnehmern die Möglichkeit, nacheinander oder auch gleichzeitig mit jedem anderen gewünschten Netzteilnehmer zum Zweck des Datenaustausches in Verbindung zu treten. Realisiert wird ein **Rechnernetz** in der Regel durch eine Anzahl von **Vermittlungsknoten**, d.h. speziell für die Aufgabe der Vermittlung und Übertragung digitaler Daten entwickelten IT-Systemen und den **Verbindungen** der Knoten untereinander.

Damit die Rechnersysteme miteinander über Rechnernetze kommunizieren können, brauchen sie Kommunikationssoftware (und auch Hardware). Die Kommunikationssoftware wird in der Form eines Kommunikationsstack in Rechnersystemen auf verschiedenen Ebenen zur Verfügung gestellt.

Im Folgenden wird exemplarisch ein Bild des TCP/IP-Kommunikationsstack für Rechnersysteme dargestellt, welcher im Internet die größte Bedeutung in der Kommunikation darstellt.

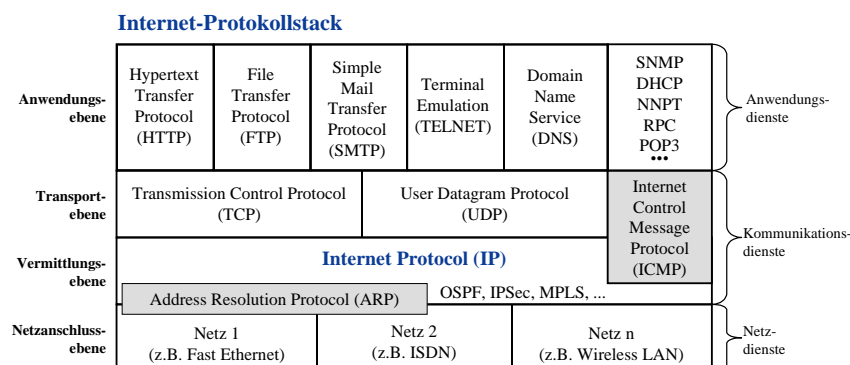


Abbildung: TCP/IP-Kommunikationsstack

Der TCP/IP-Kommunikationsstack wird in unterschiedliche Ebenen aufgeteilt.

¹⁸ Proebster, S. 33.

Netzanschlussebene

Die Netzanschlussebene ermöglicht einem Rechnersystem Daten zu einem anderen Rechnersystem innerhalb des direkt angeschlossenen Netzes (z. B. Ethernet, ISDN, Wireless-LAN...) zu übertragen. Dazu sind genaue Kenntnisse des zugrunde liegenden Netzaufbaus nötig. Die Netzzugangsebene umfasst die zwei unteren Ebenen des OSI-Modells und beinhaltet die Kapselung von IP-Paketen im Netzrahmen (Frames) und die Zuordnung von IP-Adressen zu physikalischen Netzadressen, z.B. MAC-Adressen. Die Netzanschlussebene besteht aus Kommunikationshardware (Ethernet, ISDN, usw.), auf der selber ein Teil der Kommunikationssoftware dieser Ebene realisiert sein kann, sowie der Rest der Kommunikationssoftware dieser Ebene, die i.d.R. als Treiber realisiert wird.

Vermittlungsebene

Die Vermittlungsebene definiert den Aufbau von IP-Paketen und bestimmt, auf welchem Weg die Daten durch das Internet (IP-Netz) übertragen werden (Routing). Hier muss die Anpassung an die Netzanschlussebene erfolgen. Sie besteht aus Kommunikationssoftware, die i.d.R. in das Betriebssystem integriert ist.

Transportebene

Die Transportebene stellt eine Verbindung zwischen zwei Endpunkten oder Rechnersystemen her. Die wichtigsten Protokolle sind hier TCP und UDP. Sie besteht aus Kommunikationssoftware, die i.d.R. ebenfalls in das Betriebssystem integriert ist.

Anwendungsebene

Die Anwendungsebene beinhaltet sämtliche Computerprogramme und Dienste, die über die Netzwerkverbindung durchgeführt werden sollen. Dazu gehören vor allem Dienste wie FTP (Datentransfer zwischen zwei Rechnersystemen), SMTP (E-Mail-Funktionen), HTTP (World Wide Web), usw.¹⁹. Die Anwendungsebene besteht aus Kommunikationssoftware, die i.d.R. als Anwendung oder in einer Anwendung realisiert ist.

5.2.2 Diskussion über die Abhängigkeit der Interoperabilität

Da die Notwendigkeit über Netzwerke Daten auszutauschen in unserer modernen internationalen Informationsgesellschaft immer größer wird, spielt die direkte und indirekte Interoperabilität von Kommunikationssystemen eine enorme Rolle und muss 100% gegeben sein, um eine wachsende IT-Wirtschaft zu ermöglichen.

Die Kommunikation zwischen zwei Rechnersystemen bzw. zwischen einem Rechnersystem und einem Netzknoten ist nur dann zu realisieren, wenn die Softwareschnittstellen zu dem Kommunikationsprogramm, welches die Kommunikation ausführen soll, bekannt ist.

Die Kommunikation auf den Netzwerkebenen wird mit Kommunikationsprotokollen realisiert, die i.d.R. bei IEEE, ITU-T und ISO als Norm standardisiert sind (siehe auch Kapitel Standardisierung und Normung).

¹⁹ *Tanenbaum*; siehe auch S. 58.

Die Kommunikation auf den höheren Ebenen (Vermittlungsebene, Transportebene und Anwendungsebene) wird zum größten Teil über Kommunikationsprotokolle realisiert, die bei **IETF** standardisiert sind (siehe auch Kapitel Standardisierung und Normung). Der Erfolg des Internet beruht auf der Tatsache, dass diese Kommunikationsprotokolle erstens relativ einfach zu implementieren sind und zweitens frei zu verwenden sind. Da die Protokolle aufeinander aufbauen, ist die Größe der Abhängigkeit auf der untersten Ebene am größten und auf der höchsten Ebene am kleinsten. Jedes Unternehmen ist im Prinzip in der Lage, über die "Standards" und die Referenzimplementierungen, insbesondere bei IETF, eine eigene Kommunikationssoftware zu implementieren (siehe Kap. Standardisierung und Normung).

In Kommunikationsnetzen, die üblicherweise eine heterogene Landschaft aus Komponenten verschiedenster Hersteller bilden, ist ein Höchstmaß an Interoperabilität Grundvoraussetzung für die Leistungsfähigkeit des Netzes (Internet), und damit der IT-Wirtschaft. Bis auf einige wenige proprietäre Protokolle, sind die meisten wichtigen Protokolle genormt und von daher auch gut dokumentiert sowie frei verfügbar und nutzbar. Wie mit der Verwendung eines proprietären Protokolls aus einer Monopolstellung heraus versucht werden kann einen weiteren Bereich zu monopolisieren, wird am Beispiel **CIFS** im Abschnitt "Verteilte Systeme" diskutiert. Aus diesem Grund ist die Interoperabilität in Kommunikationssystemen i.d.R. gewährleistet. Auch in Zukunft soll darauf geachtet werden, dass wichtige Kommunikationsprotokolle durch Standards gestützt werden.

Zur Bedeutung von Normen siehe Kapitel "Standardisierung und Normung" (6.4).

5.3 Verteilte Systeme

Im folgenden Abschnitt wird das Softwaresystem "Verteilte Systeme" beschrieben und die Abhängigkeit der Interoperabilität diskutiert.

5.3.1 Ziele und Aufgaben

Ein **Verteiltes System** ist eine Menge voneinander unabhängiger Rechnersysteme, die dem Benutzer wie ein einzelnes System erscheint²⁰.

Dies beinhaltet zwei Aspekte:

1. Die Rechnersysteme sind autonom (Hardware)
2. Die Benutzer haben den Eindruck, sie hätten es mit einem einzigen System zu tun (Software - virtuelle Maschine).

Ein Verteiltes System baut auf ein Rechnernetz auf. Um heterogene Rechnersysteme und Netzwerke zu unterstützen, und gleichzeitig den Eindruck eines einzigen Systems zu präsentieren, werden verteilte Systeme häufig mithilfe einer Softwareschicht organisiert. Diese **Softwareschicht (Middleware)** liegt logisch zwischen einer Ebene des Benutzers und der Applikation und der Ebene des Betriebssystems.

²⁰ Tanenbaum/Stehen, S. 18.

Verteilte Systeme arbeiteten mit einer zusätzlichen Software-Schicht, der Middleware, um die Heterogenität der vielen verschiedenen zugrunde liegenden Plattformen mehr oder weniger zu verbergen, aber auch, um sich um Verteilungsaspekte zu kümmern. Neben den Designkriterien "Benutzer und Ressourcen verbinden", "Transparenz" und "Skalierbarkeit", spielt die Offenheit bei Verteilten Systeme eine besondere Rolle.

In einem Verteilten System kann keine homogene Hardwareumgebung und einheitliche Software vorausgesetzt werden. Damit jedoch eine Zusammenarbeit stattfinden kann, müssen einige Bedingungen erfüllt sein.

- Die Schlüsselschnittstellen der Hardware- und Softwaremodule müssen offen gelegt sein.
- Die Interprozesskommunikation auf Anwendungsebene muss einheitlich und offen gelegt sein, damit verteilte Anwendungen miteinander Informationen austauschen können.

Beispiele von Verteilten Systemen sind²¹:

- Mit dem Paradigma "Verteilte Objekte"
 - Distributed COM von Microsoft
 - CORBA von Objekt Management Group (OMG)
- Mit dem Paradigma "Dateisystem"
 - NFS von SUN Mircosystems
 - CODA in Linux
- Mit dem Paradigma "dokumentbasierte System"
 - World Wide Web (WWW); Internet (ISTF und W3C)
 - Lotus Notes von IBM
- Mit dem Paradigma "koordinationsbasierte Systeme "
 - TB/Rendezvous
 - JINI von SUN Mircosystems

²¹ *Tanenbaum/Stehen, S. 554, ff.*

5.3.2 Diskussion über die Abhängigkeit der Interoperabilität

Damit die Maschinen (Rechnersysteme) A, B und C im vorherigen Bild miteinander über die Middleware-Dienste transparent Aufgaben gemeinsam lösen können, müssen alle auf die Middleware-Software zugreifen können.

Da viele Verteilte System über Interessensgemeinschaften zusammenkommen, ist ein Konsens und Interoperabilität durch verfügbare Spezifikationen mit Schnittstellenbeschreibungen i.d.R. gegeben. Andere Verteilte Systeme wie z.B. das World Wide Web (WWW) werden in Standardisierungsgremien (IETF und W3C) genormt. Es gibt aber auch Verteilte Systeme wie "Lotus Notes" von IBM und "Distributed COM" von Microsoft bei denen die Integration von Individualsoftware, um z.B. in Anwenderunternehmen Geschäftsprozesse optimal gestalten zu können, notwendig sind. Verteilte Systeme wie "Lotus Notes" und "Distributed COM" sind stark verbreitet, so dass die Integration von Individualsoftware unumgänglich sind, um die Funktionalität in Gesamtsysteme einzubinden zu können.

Aus diesem Grund ist es notwendig, dass alle Verteilten Systeme mit offenen Schnittstellen arbeiten, damit für die Zukunft der IT-Wirtschaft keine Behinderung durch die nicht zur Verfügung Stellung von Schnittstellen durch Konkurrenten auftritt.

Im Folgenden wird als ein Beispiel exemplarisch "Client - Samba-Server" dargestellt, welches aufzeigen soll, warum Reverse Engineering zwischen Client und Server notwendig ist.

Reverse Engineering als Mittel zur Erlangung der Interoperabilität zu proprietären Software Schnittstellen

CIFS, das „Common Internet File Sharing“ Protokoll von Microsoft, ist das populärste Protokoll zum Austausch von Dateien innerhalb von Local Area Networks - LANs. Die Protokolle, die für den Austausch von Dateien und den Zugriff auf Druckdienste innerhalb von lokalen Netzwerken sorgen, sind deshalb so populär, weil sie seit den frühesten Windows Versionen ein integraler Bestandteil des Betriebssystems sind.

Ein weiterer de facto Standard in der Welt der lokalen Netzwerke stellt das Windows Domänen Konzept dar. In einem homogenen Microsoft Windows Netzwerk authentifizieren sich die Arbeitsstationen an einem Domänen Controller, und können nun bestimmte Ressourcen (z.B. Drucker oder Dateifreigaben über die CIFS Protokolle) nutzen. Auch hier hat eine breite Integration in sämtliche Windows Server und Client Systeme für eine de facto Standardisierung gesorgt.

Diese Technologien erlauben in einer homogenen Microsoft Windows Umgebung den Aufbau einer Infrastruktur, die sowohl den Dateiaustausch und den Druckerzugriff über ein Netzwerk erlauben, als auch einen verteilten Authentifizierungsmechanismus bieten.

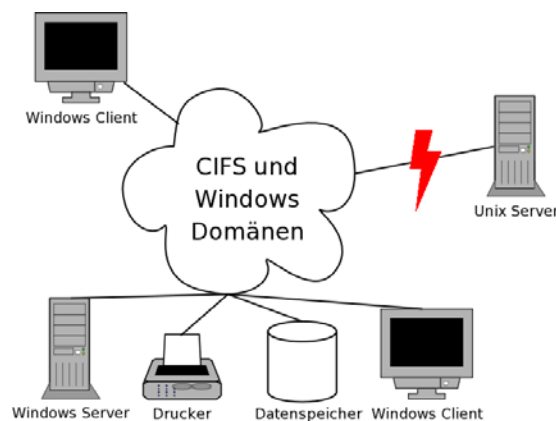


Abbildung: CIFS

Die Problematik, der sich bereits Anfang der 90er Jahre der australische Student Andrew Tridgell gegenüber sah, liegt auf der Hand: Es fehlt die Kommunikation zu und die Nutzung von allen anderen Betriebssystemen²². Da die proprietäre Softwareschnittstelle von Microsoft nicht öffentlich dokumentiert war, nahm Tridgell die Sache selbst in die Hand und analysierte die Regeln des CIFS- Unterprotokolles SMB (Server Message Block) durch Abhören des Netzwerkverkehrs und schrieb auf Grundlage dieser Informationen seine eigene Version des CIFS Protokolls. Er legte somit – ohne es zunächst selbst zu wissen – den Grundstein für den Samba²³ Server.

Einige Jahre später wurde der Samba Server weiterentwickelt und aufgrund der sehr spärlichen Informationspolitik Microsoft's war das einzig probate Mittel, das CIFS Protokoll weiter als Open Source zu implementieren, das „Reverse Engineering“²⁴, also durch die Analyse des Netzwerkverkehrs auf die reale Umsetzung zu schließen. Durch den Einsatz von vielen Entwicklern und der Open Source Community wurde Samba schnell zu einer ernstzunehmenden Alternative zu den Windows File- und Printservern, insbesondere, da die Arbeitsstationen, die auf diese Server zugreifen, den Unterschied nicht bemerken.²⁵

Heutzutage konzentriert sich das Samba Team auf die Weiterentwicklung des Servers zu einem vollwertigen Domänenkontroller, und wieder ist dies nur durch Reverse Engineering möglich, da auch hier nur wenige Dokumentationen zur Verfügung stehen. Nur durch die Technik des Reverse Engineering war es möglich, eine alternative Server Software zu den Produkten, die von Microsoft aufgrund ihrer Dominanz im Desktop-Bereich zu de facto Standards gemacht wurden, zu erstellen und die Schnittstellen des CIFS Protokolls und der Windows Domänen zu nutzen.

Wäre ein Reverse Engineering einer Schnittstelle nicht möglich, dann kann z.B. durch die Geheimhaltung von solchen Schnittstellen eine bestehende Dominanz in einem Bereich (im Beispiel: Client-Bereich) genutzt werden, um eine Dominanz in einem weiteren Bereich (im Beispiel: Server-Bereich) zu erlangen. Damit hätte der Inhaber eines Monopols die Möglichkeit, dieses auf einen anderen Marktsektor auszudehnen, auf dem er nicht notwendigerweise selbst wirtschaftlich aktiv ist.

²² http://www.linux-mag.com/1999-09/samba_01.html.

²³ <http://de.samba.org>.

²⁴ http://de.wikipedia.org/wiki/Reverse_engineering.

²⁵ <http://www.ubiqx.org/cifs/Intro.html>.

Hintergrund

In vielen Unternehmen kommt Windows auf den Workstation zum Einsatz, und UNIX ist häufiger im Serverbereich anzutreffen. Die in jeder Workstation das Modul für CIFS bereits integriert ist, nicht jedoch alternative Systeme wie z.B. NFS, ist es nahe liegend, das UNIX-Server lieber mit CIFS ausgerüstet werden.

5.4 Anwendungssysteme

Im folgenden Abschnitt wird das Softwaresystem "Anwendung" beschrieben und die Abhängigkeit der Interoperabilität diskutiert.

5.4.1 Ziele und Aufgaben

Ein Anwendungssystem ist ein System, das Softwaremodule enthält. Im weiteren Sinne umfasst es eine Menge von inhaltlich zusammengehörigen Aufgaben (Funktionen, Operationen), die dafür verantwortlichen Menschen als Aufgabenträger und die zu ihrer Erfüllung eingesetzte technische Ausstattung. Im engeren Sinn wird darunter oft ein Anwendungsprogramm (d.h. das reine Softwaresystem) verstanden, das eine spezifische Aufgabe unterstützt.²⁶

Anwendungsprogramme sind z.B. allgemeine Textverarbeitungsprogramme, Tabellenkalkulationen, Präsentationssysteme, Graphikprogramme, Computerspiele, usw. aber auch ERP (Enterprise Resource Planning Systeme), kundenspezifische Individuallösungen, Software zur Steuerung von Maschinen, Kommunikationssoftware in Telefonen und Vermittlungen oder auch Software in Geldautomaten, usw.

Im Office-Bereich ist der Wunsch der Anwender besonders groß, dass die Interoperabilität von Austauschformaten vorhanden ist. In der Praxis ist es schwer vorstellbar, dass viele Officeformate parallel existieren und nicht miteinander interoperabel sind, oder mit Hilfe von Konvertierungssystemen interoperabel gemacht werden können!

5.4.2 Diskussion über die Abhängigkeit der Interoperabilität

Falls Anwendungen nur auf einem Rechnersystem laufen und der Input und Output nicht von anderen Anwendungen abhängig sind, kann die Interoperabilität vernachlässigt werden, da Anwendungen dann i.d.R. nur auf Standardfunktionen des Betriebssystem zugreifen und eine Interaktion mit anderen Softwaresystemen nicht notwendig ist, wie z.B. bei einem elektronischen Lexikon.

Bei einzelne Funktionen in Anwendungen wie z.B. der "1-Click-Buy" oder "Fortschrittsbalken" sind zwar aus Marketinggesichtspunkten und Benutzerkomfort sehr wichtig, spielen aller für eine zur Verfügung Stellung einer Gesamtfunktionalität (z.B. eine spezielle Anwendung), und damit für die Interoperabilität, eher eine untergeordnete Rolle.

Wenn jedoch die Lösung eines Gesamtproblems auf verschiedene Anwendung verteilt ist oder der Input bzw. der Output von anderen Anwendungen abhängt, spielt die Interoperabilität ein sehr wichtige Rolle. Aus diesem Grund ist die Interoperabilität z.B. von Datenformaten (DOC,

²⁶ <http://www.tfh-berlin.de/~sheshonk/GIAK/defs.html#Anwendungssystem>.

RTF, PDF, usw.) und Datenstrukturen (XML, SGML, usw.) für die Integration verschiedener Anwendungen und deren Ergebnisse in ein Gesamtsystem von hoher Bedeutung für die Erbringung zusätzlicher Funktionalität. Gerade im Bereich der Individualsoftwareentwicklung in speziellen Branchen trifft dieser Fall besonders zu.

Die Interoperabilität zwischen verschiedenen Rechnersystemen bzw. Softwaresystemen ist die Schwerpunkt der Optimierung von Geschäftsprozessen. Dabei müssen Daten in unterschiedlichen Formaten von unterschiedlichen Rechnersystemen geholt, verarbeiten und wieder zurückgegeben werden. Evtl. müssen bei diesen Aktionen entsprechende Konvertierungen durchgeführt werden (siehe Kap. "Konvertierungsprogramme, Nachrichten-Broker").

Weitere Aspekte

Ein weiterer wichtiger Punkt bei der Betrachtung der Interoperabilität ist, dass ein sehr großer Anteil von Software nicht in Form von vermarktungsfähigen Standardprodukten, sondern als Individualsoftware direkt in Anwenderunternehmen zur Gestaltung ihrer Geschäftsprozesse geschrieben wird. Unternehmen, die solche Individualsoftware herstellen, sind darauf angewiesen, dass die von ihnen erstellen Softwareprodukte in der Lage sind, mit kommerziellen Softwareprodukten (Microsoft, SAP, ...) interagieren zu können. Falls die Softwareschnittstellen nicht bekannt sind, kann faktisch keine Individualsoftware erstellt werden, die auf der Basis von Standardsoftwareprodukten arbeitet.

Aus diesem Grund ist die Interoperabilität mit Standardanwendungssoftwareprodukten enorm wichtig, um die notwendige Integration von verschiedenen Anwendungen und die Weiterverarbeitung von Ergebnissen zu ermöglichen. Falls die Interoperabilität mit Standardsoftwareprodukten durch die großen Hersteller verhindert wird, werden die kleineren Hersteller, die sich in der Regel durch die Nähe zu den Anwendungsbranchen profilieren, aus dem Softwaremarkt verdrängt und die notwendige Individualsoftware kann nicht oder wesentlich erschwert integriert werden, was den IT-Markt in Deutschland deutlich reduziert wird (siehe Kap. Softwaremarkt: Mehr als 50 % der Unternehmen machen individuelle Anpassungen von Software).

5.5 Trusted Computing, Nexus

Im folgenden Abschnitt wird das Softwaresystem "Trusted Computing" als ein neues Sicherheitskonzept beschrieben und die Abhängigkeit der Interoperabilität diskutiert.

5.5.1 Ziele und Aufgaben

Das Thema Trusted Computing kann für unsere Zwecke grob in zwei Teile unterteilt werden. Erstens in die Basistechnologien, die z.B. von der Trusted Computing Group zur Verfügung gestellt wird, und in Nexus, einer neuen Softwareschicht in einem Betriebssystem.

Trusted Computing Group

Die Sicherheit von Plattformen sind eine Möglichkeit, sich gegenseitig als vertrauenswürdig zu authentisieren. Sie soll in kommenden Rechnersystemen nach den Vorstellungen der Hersteller von Informationstechnik in einem neuen Hardwarebaustein verankert werden. Dieser Baustein wird von einer Gruppe von Unternehmen (die meisten sind große US-Unternehmen), die sich

als "Trusted Computing Group" (TCG)²⁷ zusammengeschlossen haben, unter dem Namen "Trusted Platform Module" (TPM) spezifiziert. Die zum großen Teil frei zugänglichen Spezifikationen werden von der TCG laufend überarbeitet und aktualisiert. Erste Versionen der Spezifikation dienen bereits heute als Grundlage für die millionenfache Produktion von TPM-Chips und deren Integration in Informationstechnik. Von den kryptographischen Grundlagen bis hin zum praktischen Einsatz wird die breite Einführung des TPM erhebliche Auswirkungen auf den Umgang mit digitalen Informationen haben²⁸.

Software Stack (TSS) Specification

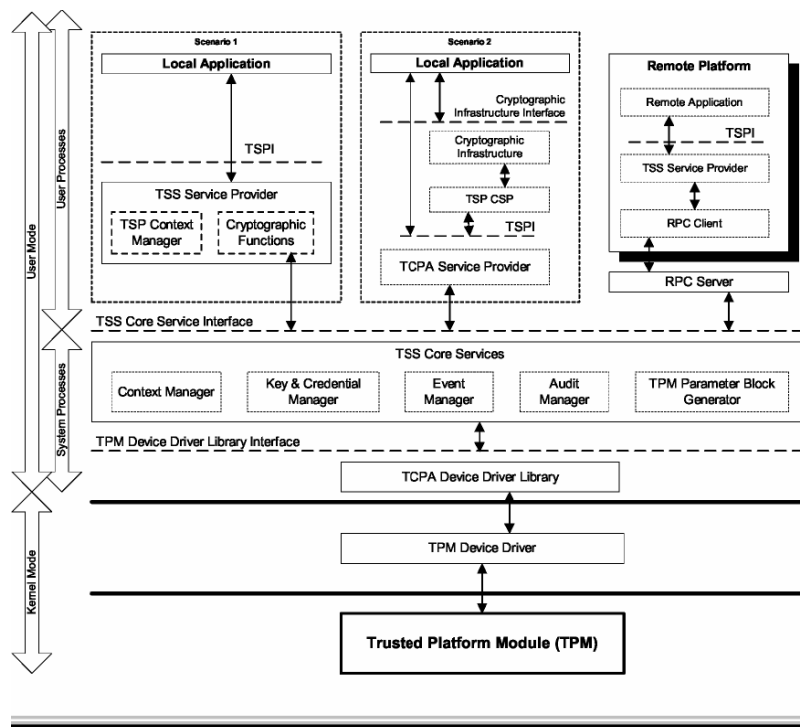


Figure 1-2 TSS Architecture Diagram

Die aktuellen Beschreibungen zu den Funktionen und Verfahren sind in den folgenden Papieren beschrieben²⁹:

TCG TPM Specification Version 1.2³¹

- Design Principles
- Structures of the TPM
- TPM Commands

²⁷ <https://www.trustedcomputinggroup.org/home>.

²⁸ siehe <http://www.bsi.de/trustcomp/>.

²⁹ <https://www.trustedcomputinggroup.org/home>; Design Principles, Structures of the TPM und TPM Commands.

³¹ <https://www.trustedcomputinggroup.org/downloads/specifications/>.

Information über den Umgang mit Rechten in der TCG³²

Immer dann, wenn eine Spezifikation fertig gestellt ist, wird vor der Veröffentlichung diese Spezifikation allen Mitgliedern zur Verfügung gestellt und eine Zeit von 60 Tagen abgewartet, in den die Mitglieder besondere Anforderungen bezüglich IPR anzeigen können. Falls in dieser Zeit keine besonderen Anforderungen angezeigt werden, ist automatisch vereinbart, dass zwischen den Mitgliedern der TCG RAND Konditionen vereinbart sind.

Hintergrund dieser Vereinbarung ist, dass jedes der Mitglieder faire Konditionen von den anderen Mitgliedern erwarten kann. Falls jemand in TCG eintritt, um diese Konditionen nutzen zu können, muss er im Gegenzug das gleiche Recht auch mit seinen IPRs einräumen.

Hinweis:

Im Bereich Trusted und Secure Computing gibt es in den USA mehrere 100 Patente. Ein Patent ist z.B. von IBM: "Method and system for updating a root of trust measurement function in a personal computer".³³ Ein anderes Patent ist z.B. von Intel: "Secure transfer of trust in a computing system".³⁴

Typisches Verhalten von großen Unternehmen im Bereich von Rechtemanagement:

Patente werden solange zurückgehalten, wie die "patentierete Technik" schon in der Anwendungen ist. Damit werden zwei Aspekte erreicht:

1. Dadurch dass die "patentierete Technologie" schon im Einsatz ist, wird auf jeden Fall eine Patentgebühr fällig, und die Wahrscheinlichkeit weiterer Patentlizenzen ist sehr groß, da in der Praxis in nachhinein die "patentierete Technologie" nicht so einfach umgangen werden kann.
2. Das im Standard enthaltende Patent, bleibt erhalten, der Standard hat sich schon verbreitet und eine Änderung des Standards ist sehr unwahrscheinlich.

Hinweis:

Die Einschränkung, dass dann "nur" noch unter RAND Patengebühren erzielt werden können ist dann zu vernachlässigen.

Next-Generation Secure Computing Base (NGSCB) / Nexus

NGSCB (Next-Generation Secure Computing Base) ist der Oberbegriff für eine Reihe von Sicherheitsfunktionen in zukünftigen Windows-Versionen. Das neue Sicherheitskonzept basiert

³² <https://www.trustedcomputinggroup.org/home>; Trusted Computing Group Bylaws.

³³ US Pat.No.6,782,349 zuletzt abgerufen am 4. Oktober 2004 , <http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO2&Sect2=HITOFF&p=1&u=/netahtml/search-bool.html&r=1&f=G&l=50&co1=AND&d=ptxt&s1='updating+root'.ABST.&s2='personal+computer'&OS=ABST/>.

³⁴ US Pat.No.6,539,480 zuletzt abgerufen am 4. Oktober 2004 <http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO2&Sect2=HITOFF&p=1&u=/netahtml/search-bool.html&r=38&f=G&l=50&co1=AND&d=ptxt&s1='Secure+Computing'&OS='Secure+Computing'&RS='Secure+Computing'>.

auf neuen Software-Sicherheitsfeatures, die mit hardwarebasierten Funktionen kombiniert werden. Als sicheren Vertrauensanker für die neuen NGSCB-fähigen Windows-Versionen wird Microsoft eine Reihe der von TCG spezifizierten Funktionen eines Smartcard-ähnlichen Chips, dem sog. Trusted Platform Module (TPM), unterstützen, der zukünftig fest auf der Hauptplatine von NGSCB-fähigen Rechnersystemen aufgebracht sein wird. NGSCB unterstützt vier Grundfunktionen:

- Neue Prozessorfunktionen und Chipsatz ermöglichen sichere Abschottung von Bereichen des Arbeitsspeichers (geschützter Speicherbereich = curtained memory)
- Kleiner, überschaubarer, in Windows integrierter Sicherheitskern (Nexus) kontrolliert einen neuen Trusted Mode. Software kann unter der NGSCB "hinter der Mauer" im Trusted Mode (sog. "Nexus Computing Agents (NCA)") besonders sicher ausgeführt werden: Weder Schadprogramme (Viren, Trojaner etc.) noch andere Applikationen oder Betriebssystemkomponenten können NCAs beeinträchtigen.
- NGSCB erlaubt den sicheren und verifizierbaren Aufbau vertrauensvoller Kommunikationsbeziehungen zwischen PCs (Attestation).

Beispiele: Sicherer Aufbau virtueller privater Netze oder E-Banking-Applikationen

- Sensible Informationen können Applikationen und/oder der Plattform eindeutig und sicher zugeordnet werden (Sealed Storage = PC als "Datensafe"). Sensible Daten und Geheimnisse (Passwörter, Schlüssel, Systemzugangsinformationen) können so auf dem PC sehr zuverlässig geschützt werden.
- Anwender und Peripherie-Geräte können mit dem System über sichere Ein- und Ausgabekanäle kommunizieren (Trusted I/O). Dies ermöglicht lückenloses Vertrauen von der Tastatur und der Maus bis zum Bildschirm, z.B. zur sicheren Erzeugung elektronischer Signaturen³⁵.

Hinweis:

Neben Microsoft arbeiten noch andere Firmen und Hochschulen an solchen Realisierungen von Secure Computing Base.

- EMSCB (European Multilateral-Secure Computing Base): Uni Bochum³⁶
- LaGrande: Intel³⁷

³⁵ [http://www.competence-site.de/itsecurity.nsf/news/483FE1495D515E5CC1256DFE00632B6F/\\$file/VR_IT_Sicherheit_Frage_5.pdf](http://www.competence-site.de/itsecurity.nsf/news/483FE1495D515E5CC1256DFE00632B6F/$file/VR_IT_Sicherheit_Frage_5.pdf).

³⁶ *Sadeghi/Stüble*, Artikel Open Nexus.

³⁷ Intel 2003, White Paper zu LaGrande.

5.5.2 Diskussion über die Abhängigkeit der Interoperabilität

Da der Sicherheitskern (Nexus) als "unterstes" Softwaremodul die Hardware und die gesamte Software kontrolliert, kommt diesem Softwaremodul und dem dazugehörigen Sicherheitshardwaremodul (TPM) bezüglich der Interoperabilität eine besondere wichtige Rolle zu.

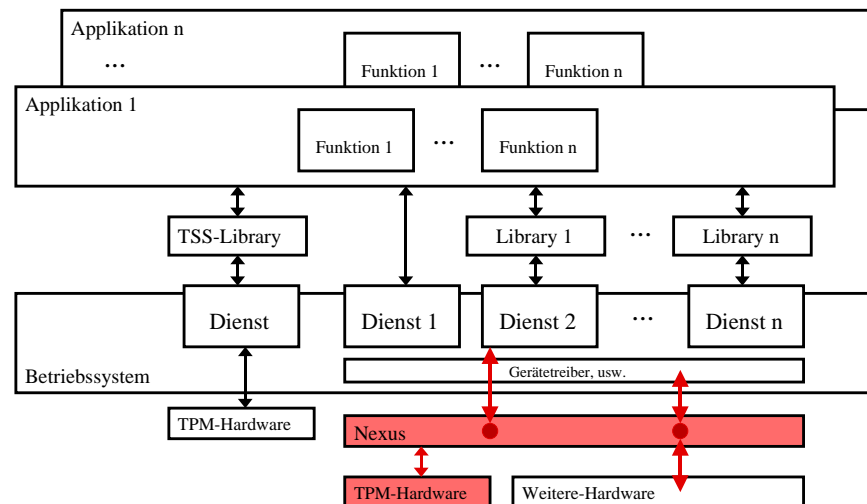


Abbildung: Nexus

Nur wer in der Lage ist, dieses Softwaremodul durch einen entsprechenden Input (Security Policy) zu steuern, kann bestimmen, welche anderen Softwaremodule auf dem Rechnersystem welche weiteren Ressourcen nutzen dürfen. Wenn dies der Benutzer eines Rechnersystems ist, kann mit einer Security Policy für einen gewollten und kontrollierten Ablauf gesorgt werden und damit das Ziel von mehr Vertrauenswürdigkeit in Rechnersystemen erreicht werden. Problematisch wird es nur, wenn die Softwareschnittstelle, mit der dazugehörigen kryptographischen Basis, nicht von dem Benutzer genutzt werden kann. Im Fall von Nexus und TPM ist daher zwingend auch die Offenlegung des kryptographischen Konzepts notwendig, damit Benutzer in der Lage sind ihre eigene Security Policy zu erzeugen.

Aus diesem Grund ist die Interoperabilität zu einem Sicherheitskern (Nexus) und zu den TPM-Funktionen unbedingt erforderlich.

Digital Rights Management (DRM)

Digital Rights Management Systeme bestehen aus einer Verknüpfung von technischen Komponenten (TPM) und einer speziellen Anwendungsebene (z.B. NGSCB). DRM-Systeme verwirklichen die Idee der zentralen Kontrolle digitaler Inhalte durch kryptographische Verfahren. Realisiert wird dies, indem ein beliebiger digitaler Inhalt eindeutig an ein beliebiges Gerät und/oder Datenträger auf einzigartige Weise kryptographisch gebunden wird. Ohne den entsprechenden Schlüssel für den digitalen Inhalt vom Rechteinhaber ausgehändigt zu bekommen, kann der Benutzer zwar das Gerät oder den Datenträger erwerben - jedoch nicht auf den Inhalt zugreifen. DRM-Systeme werden zur Wahrung und Durchsetzung von Wünschen eines Rechteinhabers technisch so konzipiert, dass ein Schlüsselaustausch ohne Einflussnahme des Benutzers

für jedes einzelne Gerät vorgenommen werden kann. Dies ermöglicht den Rechteinhabern neue Vertriebsformen wie z.B. eine zeitweise Vermietung des Inhalts³⁸.

Solche DRM-Systeme können von Firmen für eine interne Security-Policy aber auch für die Gewährleistung externer Security-Policies genutzt werden.

Die Interoperabilität spielt bei Digital Rights Management (DRM) eine besondere Rolle. Auch hier müssen die Risiken von nicht verfügbaren Schnittstellen besonderes betrachtet werden.

Beispiel eines möglichen Szenario:

- Große Unternehmen stellen ihren Kunden Rechnersysteme (PC, Notebook, PDAs, usw.) quasi kostenlos zur Verfügung.
- Diese Unternehmen verdienen dann ihr Geld dadurch, dass sie die Software und Informationen mit der Hilfe von DRM verkaufen, d.h. es kann genau bestimmt werden, auf welchen Rechnersystemen und wie oft eine Software ablaufen kann.
- Dadurch können sie den Markt der Software für Rechnersysteme für sich allein gestalten, und haben die Möglichkeit alle anderen auszuschließen.

Bewertung und Maßnahmen:

Da DRM-Systeme nicht nur mit offenen Schnittstellen arbeiten, sondern auch den Zugang zu Kryptosystemen ermöglichen müssen, ist hier bezüglich des Wettbewerbs besondere Aufmerksamkeit notwendig.

Hinweis:

Im Bereich Digital Rights Management gibt es in den USA viele Patente. Ein Patent ist z.B. von Microsoft: "Digital rights management operating system".³⁹

5.6 Kryptoalgorithmen, Sicherheitsprotokolle und Sicherheitsinfrastrukturen

Im folgenden Abschnitt werden die Softwaresysteme "Kryptoalgorithmen", "Sicherheitsprotokolle" und "Sicherheitsinfrastrukturen" als ein besonders sensibler Bereich von Softwaresystemen beschrieben und die Abhängigkeit der Interoperabilität diskutiert.

³⁸ http://de.wikipedia.org/wiki/Digital_Rights_Management.

³⁹ <http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=/netahtml/srchnum.htm&r=1&f=G&l=50&s1=6,330,670.WKU.&OS=PN/6,330,670&RS=PN/6,330,670>.

5.6.1 Ziele und Aufgaben

Kryptoalgorithmen dienen z.B. dazu, Daten zu verschlüsseln, digitale Signaturen durchzuführen, Schlüssel vertraulich auszutauschen und Hashwerte zu berechnen⁴⁰.

Beispiele von Kryptoalgorithmen für die Verschlüsselung von Daten sind:

- Data Encryption Standard (DES)
- IDEA-Algorithmen
- Blowfish-Algorithmen
- Advanced Encryption Standard (AES)
- usw.

Beispiele von Kryptoalgorithmen für die digitale Signaturen sind:

- RSA-Verfahren
- Elliptische Kurven
- usw.

Beispiele von Kryptoalgorithmen für den Austausch von geheimen Schlüssel sind:

- RSA-Verfahren
- Diffie-Hellman Verfahren
- usw.

Beispiele von Kryptoalgorithmen für die Hashwert-Berechnung sind:

- Message Authentication Code (MAC)
- MD5
- SHA mit mehreren Varianten
- RIPEMD
- usw.

⁴⁰ *Blumberg/Pohlmann*, Kapitel 4.4.

Sicherheitsprotokolle dienen dazu mit Hilfe von Protokollen die Kryptographiealgorithmen auszuhandeln, Session-Schlüssel zu verarbeiten, Modes of Operations zu definieren und weitere Modies, die z.B. die Authentifikation betreffen, auszuhandeln.

Beispiele von Sicherheitsprotokollen sind:

- IPSec
- SSL / TLS
- usw.

Sicherheitsinfrastrukturen dienen dazu das Key-Management zu organisieren.

Beispiele von Sicherheitsinfrastrukturen sind:

- Key-Management-Systeme auf der Basis von symmetrischen Verschlüsselungsverfahren
- Public-Key-Infrastrukturen auf der Basis von asymmetrischen Verfahren
- usw.

5.6.2 Diskussion über die Abhängigkeit der Interoperabilität

Die Situation bezüglich Patenten ist sehr unterschiedlich. Bei der Auswahl des Advanced Encryption Standard (AES) war ein besonderes Kriterium, dass der Algorithmen frei von Rechten sein muss. Im Bereich von "Elliptische Kurven", die bei der Nutzung von TPMs und SmartCard eine besondere Rolle spielen, gibt es das US-Unternehmen Certicom, das nach eigenen Angaben über 300 Patenten im Bereich der Implementierung von elliptischen Kurven verfügt und für die Rechte einer nicht exklusiven Nutzung 25 Mio. US\$ im Oktober 2003 von der NSA bekommen hat⁴¹.

Die Interoperabilität von Kryptoalgorithmen, Sicherheitsprotokolle und den dazugehörigen Sicherheitsinfrastrukturen ist von großer Bedeutung. Nur wenn diese zueinander passen und ein freier Zugang möglich ist, kann auf der Basis von Sicherheitsmechanismen, die mit kryptographischen Verfahren arbeiten, Interoperabilität erreicht werden, auch wenn alle anderen Softwaresysteme interoperabel wären.

Aus diesem Grund muss die Interoperabilität von Kryptoalgorithmen, Sicherheitsprotokollen und Sicherheitsinfrastrukturen gewährleistet werden. Nur dann ist der notwendige Austausch von Daten möglich (siehe dazu auch Kapitel "Testsysteme für die Interoperabilität").

Das Ziel des Einsatzes von kryptographischen Verfahren ist, dass sich bestimmte Benutzergruppen (Anwender) auf der Basis von verbreiteter Basistechnologie bezüglich der Vertraulichkeit, Verbindlichkeit und Integrität, gegenüber anderen Benutzergruppen (Konkurrenz, Hacker,

⁴¹ http://www.certicom.com/index.php?action=company,press_archive&view=175

usw.) abgrenzen können, um Geschäftsprozesse vertrauenswürdig realisieren zu können. Diese notwendigen Sicherheitsbedürfnisse dürfen nicht mit Hilfe von Patentrechten eingeschränkt werden, da sie als Enabler für die Umsetzung von elektronischen Geschäftsprozessen der Informationsgesellschaft eine absolutes Muss darstellen. Sie müssen von allen unter fairen und nicht diskriminierenden Bedingungen verwendet werden dürfen.

Mit Hilfe von kryptographischen Verfahren kann aber auch gezielt von großen Herstellern ein Ausschluss von Märkten betrieben werden, was IT-politisch bewertet werden muss, weil dadurch eine Blockierung von Märkten stattfinden kann.

Ähnlich wie die interne Sicherheit eines Landes einen besonderen Stellenwert hat, muss überprüft werden, ob die Kryptoalgorithmen und Sicherheitsprotokolle einen besonderen Stellenwert benötigen.

Hinweis: Siehe dazu auch Kap. "Standardisierung und Normung" - Auszug von IETF (6.4).

5.7 Embedded Systems

Im folgenden Abschnitt wird das Softwaresystem "Embedded System" beschrieben und die Abhängigkeit der Interoperabilität diskutiert.

5.7.1 Ziele und Aufgaben

Embedded Systems sind vollständige individuell programmierbare Computer einschließlich Speicher, Schnittstellen und Betriebssystem, die die Steuerung dezentraler Funktionseinheiten vollständig übernehmen können. Die Entwicklung geht dahin, dass solche Systeme auf einem einzigen Chip von wenigen Millimetern Kantenlänge Platz finden. Chips dieser Art lassen sich grundsätzlich in beliebige Geräte einbauen: Heizungen, Brandmelder, Schlösser, Waschmaschinen oder auch Gerätegenerationen, die es heute noch gar nicht gibt. Bei letzterem wird von so genannten "Things that think" gesprochen, also von Dingen, die denken. Das "Denken" besteht darin, komplizierte Prozesse softwaregestützt bearbeiten zu können: So gibt es beispielsweise in der Sicherheitstechnik bereits "denkende" Bewegungsmelder, die zuverlässig einen Menschen von einem Haustier unterscheiden können. Solche verteilten intelligenten Systeme, die überall in einem Gebäude und seinen Installationen integriert werden könnten, werden zunehmend über drahtlose Kommunikationsschnittstellen mobil und – beispielsweise mittels Satellitennavigation auf der Basis von GPS – "spatial a-ware" gemacht. Dies bedeutet, dass die „Things that think“ zusätzlich mit einem Bewegungsbewusstsein ausgestattet sind und damit "wissen", wo sie gerade sind oder was um sie herum gerade vorgeht⁴².

Eine Besonderheit von Embedded System ist, dass die Software im Chip gespeichert ist und nur dann ausgelesen werden kann, wenn der Hersteller dafür das Verfahren bekannt gibt. Falls der Hersteller das Verfahren nicht bekannt gibt, kann die Software praktisch nicht ausgelesen und auch kein Reverse Engineering durchgeführt werden.

⁴² siehe Studie: Anforderung an die Technologie- und Wirtschaftspolitik durch die Konvergenz der elektronischen Medien; S. 38, BWMA.

5.7.2 Diskussion über die Abhängigkeit der Interoperabilität

Um Embedded Systeme in andere Softwaresysteme einbauen zu können, werden die Schnittstellenbeschreibung zum Embedded System benötigt. Falls diese nicht bekannt ist, kann die Einbindung nicht erfolgen.

Für Embedded Systeme gibt es mit Prinzip zwei Varianten: eine offene Variante, die es ermöglicht eigene weitere Software auf das Embedded System zu laden und so die Erweiterung von eigenen Funktionen und die Integration in ein anderes Softwaresystem einfach zu bewerkstelligen.

Die geschlossene Variante kann nicht genutzt werden, wenn keine Schnittstellenbeschreibung vorliegt. Da bei geschlossenen Varianten der Embedded Systeme in der Praxis auch kein Reverse Engineering möglich ist, kann nur der Hersteller über eine geeignete Schnittstellenbeschreibung für die Interoperabilität sorgen.

Aus diesem Grund sollte für wichtige Lösungen auf der Basis von geschlossenen Embedded Systemen versucht werden, diese nur mit genormten Schnittstellen zu verwenden.

Im übrigen sind TPMs (Trusted Computation) auch Embedded Systeme.

5.8 Zusammenfassung - Softwaresysteme und Interoperabilität

In dem vorigen Kapitel sind exemplarisch nur für einige Beispiele die Abhängigkeit bezüglich der Interoperabilität diskutiert worden. Das Thema ist sehr komplex und vielfältig und es gibt sicherlich viele weitere Beispiele, bei denen andere Kriterien eine Rolle spielen. Dennoch können die beschriebenen Systeme als Grundlage verwendet werden, die den Grad der Abhängigkeit bezüglich der Interoperabilität begründen.

Wenn von wichtigen Basissoftwaresystemen/Massenprodukten (z.B. von Microsoft und SAP) Ergebnisse nicht weiterverarbeitet werden können oder das Wissen über Syntax, Semantik, Laufzeitverhalten und Fehlverhalten der Schnittstelle nicht vorhanden ist, kann eine Software-schnittstelle, nicht von einer eigenen Software genutzt werden. Dies kann zu enormen Innovationshemmnissen in der IT-Wirtschaft führen, da insbesondere Individualentwicklung von Software auf Basissoftwaresystemen verhindert werden kann.

Die Integration von Softwaremodulen in andere Softwaresysteme ist ein wesentlicher Anteil der Softwareentwicklungstätigkeit vieler Firmen. Dabei ist ein zentrales Anliegen von Anwendungsfirmen, die Integration von vorhandenen (Insel-)Lösungen zu einheitlichen Gesamtlösungen, damit die Wirtschaftlichkeit der Software für die unternehmerische Tätigkeit ausgeschöpft werden kann. Hier kommt der Interoperabilität, insbesondere von Datenaustauschformaten, eine immense Rolle zu. 76 % der deutschen Firmen, die Software entwickeln, bauen ihre Systeme auf Basissoftware auf⁴³, mit der sie interoperabel sein müssen.

Ohne die Interoperabilität zu Basissoftwaresystemen werden die Firmen, die Individualsoftware auf dieser Basis aufbauen, ihre Lösungen nicht mehr entwickeln können.

Schnittstellen stellen die Basis von Softwaresystemen dar. Aus diesem Grund sollten Schnittstellen grundsätzlich nicht schützbar sein, da sonst eine Marktbeherrschung möglich ist, die den Wettbewerb blockieren kann.

⁴³ Vgl. http://www.isi.fhg.de/iuk/dokumente/evasoft_abschlussbericht.pdf, S. 90.

Wäre ein Reverse Engineering in Zukunft nicht mehr erlaubt, dann hätten Marktführer in bestimmten Bereichen wie Microsoft, Orakel, SAP die prinzipielle Möglichkeit, den kompletten Software Markt unter sich zu beherrschen, und Firmen, die Individualsoftware entwickeln, wären nicht mehr in der Lage, um Standardsoftwareprodukte herum, ihre Dienstleistung ohne die Zustimmung der großen Marktführer im Bereich Basissoftwaresysteme/Massenprodukte anzubieten.

Denn Marktführer könnten (und würden) sich dann sehr genau überlegen, welche Schnittstellen generell publiziert werden, welche nur gegen Zahlung einer Lizenzgebühr und welche gar nicht.

Bedenkt man noch, dass diese Marktführer selbst ganz genau wissen, welche Schnittstellen für den Markt bzw. für gewisse Branchen interessant sind, dann ist es durchaus vorstellbar, dass ganz gezielt spezielle Schnittstellen undokumentiert gelassen werden.

Damit hätten die Marktführer in der Endkonsequenz sämtliche anderen Software-Hersteller in der Hand, welche um ihre Produkte herum Zusatzsoftware anbieten.

Am besten wäre, wenn Softwareschnittstellen auf Standards beruhen würden. Auf jeden Fall ist es für die Wettbewerbsfähigkeit wichtig, dass die Interoperabilität gewährleistet werden kann.

Im Folgenden wird versucht, eine Abschätzung der direkten und indirekten Abhängigkeit eines Softwaremoduls bezüglich der unterschiedlichen Softwaresysteme zu beschreiben. Da das Thema, wie gezeigt wurde, sehr komplex ist, kann diese Übersicht nur eine generelle Idee darstellen. "direkt" bedeutet, dass das Softwaremodul direkt auf das Softwaresystem zugreift und "indirekt" bedeutet, dass das Softwaremodul über eine Kette von weiteren Softwaremodulen auf das Softwaresystem zugreift

Art der Software	Abhängigkeit eines Softwaremoduls zu anderer Softwaresystemen (direkt und indirekt)
Betriebssystem	100%
Trusted Computing, Nexus	100% falls die Sicherheit notwendig ist
Krypto- und Sicherheitssysteme	100% falls die Sicherheit notwendig ist
Kommunikationssysteme	100% falls eine Kommunikation notwendig ist
Verteilte Systeme	30 bis 70% in Abhängigkeit in wieweit Funktionen und Daten für das Gesamtsystem entscheidend sind
Anwendungssysteme	20 bis 80% in Abhängigkeit in wieweit Funktionen und Daten für das Gesamtsystem entscheidend sind

Interoperabilität im Behördenbereich

Einige Initiativen im Behördenbereich, z.B. wie die "Interoperable Erbringung europaweiter elektronischer Behördendienste (eGovernment-Dienste) für öffentliche Verwaltung, Unternehmen und Bürger (IDABC)" zeigen, dass das Thema Interoperabilität wichtig ist, damit die Voraussetzung für die Wettbewerbsfähigkeit der Industrie in der Europäischen Gemeinschaft gewährleistet sind⁴⁴.

Eine weitere Initiative ist "OSCI - Transport 1.2", in der folgendes beschrieben ist: "OSCI löst die Forderung des eGovernment nach einer interoperablen, nachweisbaren, unabstreitbaren und gesicherten Kommunikation zwischen technischen Systemen auf der Grundlage von Internetstandards⁴⁵".

Hier spielt insbesondere die Wirtschaftlichkeit von Gesamtsystemen eine besondere Rolle, da die Festlegung von bestimmten Schnittstellen eine Optimierung von notwendigen Softwaretechnologien bewirken soll.

⁴⁴ Vgl. http://europa.eu.int/eur-lex/pri/en/oj/dat/2004/l_181/l_18120040518en00250035.pdf.

⁴⁵ <http://www.osci.de>.

Betroffenheitscluster des deutschen Softwaremarktes

Von der Notwendigkeit der Interoperabilität ist im Prinzip jedes Softwaremodul/Softwaresystem abhängig.

Bei der Art der Softwareentwicklung in Deutschland überwiegt in der Gesamtbetrachtung derzeit die individuelle Anpassung von Software gegenüber der originären Entwicklung:

- 14.700 Unternehmen kaufen Basis-Software bei externen Anbietern und entwickeln diese intern weiter.
- 11.700 Unternehmen entwickeln Software originär⁴⁶.

Das bedeutet, dass der größte Teil der Unternehmen von der Interoperabilität der Basis-Software abhängt.

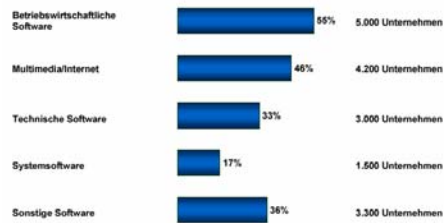


Aus Sicht der Anbieter sind die zwei zentralen wettbewerbsbestimmenden Qualitätseigenschaften von Software Zuverlässigkeit und nutzergerechte Funktionalität. Ergonomie und Benutzerfreundlichkeit sowie individuelle Lösungen sind weitere zentrale Erfolgsfaktoren. Das Angebot individueller Lösungen ist aus Sicht der Unternehmen der Sekundärbranchen sogar der bedeutendste Erfolgsfaktor. Dies verdeutlicht auch, welcher Stellenwert der individuellen Anpassung von (Standard-) Software derzeit zukommt. Die Forderung nach individuellen Lösungen spiegelt sich auch in der bereits beschriebenen Anbieterstruktur im Bereich spezieller Branchen-Software wider: Über die Hälfte der Softwareunternehmen bietet Branchenlösungen an. Für jede Branche gibt es nur relativ wenige, dafür aber hochspezialisierte Anbieter⁴⁷.

⁴⁶ http://www.isi.fhg.de/iuk/dokumente/evasoft_abschlussbericht.pdf, siehe Kap. 4.7.3 .

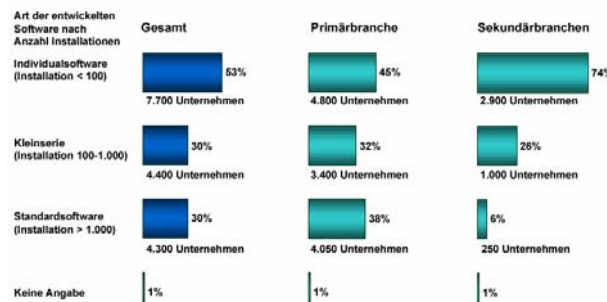
⁴⁷ http://www.isi.fhg.de/iuk/dokumente/evasoft_abschlussbericht.pdf, siehe Kap. 4.9 .

Über die Hälfte der Primärbranche bietet betriebswirtschaftliche Software als eigenständiges Produkt an - Multimedia-/Internet-Software bereits auf Rang 2



Das Kerngeschäft der Primärbranche liegt in der Entwicklung von Software, die als eigenständiges Produkt vermarktet wird. Schwerpunktmäßig wird dabei betriebswirtschaftliche Software angeboten: 5.000 Unternehmen erstellen Software, die diesem Spektrum zuzuordnen ist. Im Bereich Multimedia- und Internet-Software herrscht mit 4.200 Unternehmen die zweitgrößte Anbietervielfalt. 3.000 Unternehmen entwickeln technische Software. Systemsoftware wird von 1.500 der Unternehmen der Primärbranche angeboten. In den Bereich sonstiger (kommerzieller) Software fallen insgesamt 3.300 Unternehmen. Hierzu gehören vor allem Anbieter von Programmen für Dokumentenverwaltung oder Wissensmanagement⁴⁸.

Drei Viertel der Unternehmen der Sekundärbranchen entwickelt Individualsoftware - Standardsoftware nahezu ausschließlich durch Primärbranche entwickelt



Eine weitere Möglichkeit Software nach Kategorien zu analysieren ist die Anzahl der Installationen pro Version. Dabei lassen sich Individualsoftware (<100 Installationen), Kleinserien (100 bis 1.000 Installationen) und Standardsoftware (>1.000 Installationen) unterscheiden.

In der Gesamtbetrachtung liegt der Schwerpunkt der Softwareentwicklung in Deutschland eindeutig im Bereich der Individualsoftware. Insgesamt 7.700 Unternehmen der Primärbranche und fünf untersuchten Sekundärbranchen entwickeln Software, die weniger als 100 mal pro Version installiert wird. In den Sekundärbranchen ist dieser Schwerpunkt noch deutlicher ausgeprägt. Hier entwickeln 74% der Unternehmen Individualsoftware. Große und kleine Unternehmen der Sekundärbranchen unterscheiden sich in dieser Hinsicht nicht. Insbesondere die Softwareunternehmen im Finanzdienstleistungsbereich sind auf die Entwicklung von Individualsoftware spezialisiert.

Kleinserien (100 bis 1.000 Installationen) werden von jeweils etwa jedem dritten Unternehmen der Primär- und Sekundärbranchen angeboten.

⁴⁸ http://www.isi.fhg.de/iuk/dokumente/evasoft_abschlussbericht.pdf, siehe Kap. 4.4 .

Standardsoftware wird dagegen nahezu ausschließlich in der Primärbranche entwickelt. 4.050 der insgesamt 4.300 Unternehmen, die Standardsoftware anbieten, sind der Primärbranche zuzuordnen. Dies entspricht einem Anteil von 94%. Insbesondere Unternehmen mit 200 und mehr Mitarbeitern sind auf die Entwicklung von Standardsoftware spezialisiert.

In allen diesen Branchen spielt die Interoperabilität eine besondere Rolle.

Marktstellung der Software-Unternehmen in Deutschland

Eine internationale Spitzenstellung deutscher Unternehmen existiert bei betriebswirtschaftlicher Standard-Anwendungs-Software. Auch im Bereich der Software für Telekommunikationsnetze sind deutsche Anbieter international gut positioniert. Bei den IT-Services und der Systemintegration sind deutsche Anbieter zumindest auf dem heimischen Markt stark. Auch die Tochtergesellschaften ausländischer Konzerne erzeugen in diesen Feldern merkbare Wertschöpfung in Deutschland und auch Arbeitsplätze. Dagegen spielen deutsche Unternehmen z.B. bei den Betriebssystemen (für Mainframes, PC, PDA) und Standard-Office-Anwendungen keine bedeutende Rolle.

Unternehmen in Deutschland verfügen über Erfahrung und Know-how im Bereich Software zur Unterstützung von Geschäftsprozessen. Gerade die mittelständischen Software-Anbieter sind auf Branchenlösungen ausgerichtet⁴⁹. Wichtiges Betätigungsfeld für deutsche Anbieter ist Software für verteilte Applikationen, die damit online zugänglich und ubiquitär verfügbar werden⁵⁰.

6 Technische Voraussetzungen und Rahmenbedingungen der Herstellung von Interoperabilität

In diesem Abschnitt werden exemplarisch Beispiele beschrieben, wie eine Interoperabilität von Softwaremodulen in der Praxis erzielt werden kann.

6.1 Konvertierungsprogramme, Nachrichten-Broker

6.1.1 Beschreibung der Aufgabenstellung

Bei dem Austausch von Informationen von Rechnersystemen mit unterschiedlichen Architekturen, müssen unterschiedliche Repräsentationen für Zahlen, Zeichen und andere Datenelemente berücksichtigt werden. Unterschiede können z.B. sein: Integer: 1er oder 2er Komplement, Gleitkommazahlen, Zeichencode: z.B. ASCII oder EBCDIC, Arrays: zeilen- oder spaltenweise, niedrigstes Bit einer Zahl vorne oder hinten im Wort.

⁴⁹ siehe Studie: Anforderung an die Technologie- und Wirtschaftspolitik durch die Konvergenz der elektronischen Medien; S. 11, BWMA.

⁵⁰ siehe Studie: Anforderung an die Technologie- und Wirtschaftspolitik durch die Konvergenz der elektronischen Medien; S. 10, BWMA.

Außerdem ist es oft notwendig, Anpassung an den ausgetauschten Informationen von herstelllerspezifischen Definitionen - Formaten - durchzuführen. Standards in diesem Bereich sind z.B. HL3 (Gesundheitsbereich), XML, EDIFACT, usw.

6.1.2 Lösungen

Diese notwendige Konvertierung zwischen unterschiedlichen Formaten und Repräsentationen kann entweder als Computerprogramm in einem Rechnersystem - Konvertierungsprogramm - oder als Nachrichten-Broker - zentraler Server - realisiert werden.

Konvertierungsprogramme

Konvertierungsprogramme sind Computerprogramme, die auf einem Rechnersystem für die Konvertierung von Repräsentationen und Formaten zu sorgen. Ein Beispiel für solche Konvertierungsprogramme sind die Konvertierung von einem Graphik-Format in ein anders (GIF, TIF) oder Office-Formate (DOC, PDF, SXW) damit diese in unterschiedlichen Anwendungen verwendet werden können.

Ein weiteres Beispiel ist die Konvertierung von unterschiedlichen Film-Formaten (Digitalfernsehen, HTPC, MPEG, Media Center, PVR, DVB, Videos).

Nachrichten-Broker

Ein wichtiger Anwendungsbereich von Verteilten Systemen ist die Integration existierender und neuer Applikationen in ein einziges, kohärentes Informationssystem. Für die Integration ist es erforderlich, dass Applikationen die Nachrichten verstehen, die sie erhalten. In der Praxis ist es dafür erforderlich, dass der Sender seine ausgehenden Nachrichten im selben Format anlegt wie der Empfänger!

Das Problem bei diesem Ansatz ist, dass immer dann, wenn dem System eine Applikation hinzugefügt wird, die ein eigenes Nachrichtenformat benötigt, die Software jedes potenzieller Empfängers angepasst werden muss, um dieses Format verarbeiten zu können. Ein Alternative ist, sich auf ein gemeinsames Nachrichtenformat zu einigen, wie es bei den traditionellen Netzwerkprotokollen der Fall ist. Probleme sind dabei: unterschiedliche Abstraktionsebenen auf der Prozessebene.

Ein Nachrichten-Broker agiert als Gateway auf Applikationsebene. Die wichtigste Aufgabe von einem Nachrichten-Broker ist es, eingehende Nachrichten in ein Format umzuwandeln, das die Zielapplikationen versteht. Ein Nachrichten-Broker kann ein ganz einfacher Neumattierer für Nachrichten sein. Z.B.: eine eingehende Nachricht enthält Datenbanksätze ein anderes End-of-record-Trennzeichen als die Zielapplikation erwartet. Ein Nachrichten-Broker kann hier eine Umwandlung durchführen. Ein anderes Beispiel ist die Umwandlung von X.400 Nachrichten auf Internet-E-Mail Nachrichten. Hierbei kann nicht garantiert werden, dass alle in der eingehenden Nachricht enthaltenen Informationen umgewandelt werden können.

Im Herz eines Nachrichten-Brokers befindet sich eine Datenbank mit Regeln, die angeben, wie eine Nachricht im Format X in eine Nachricht im Format Y umgewandelt werden kann.

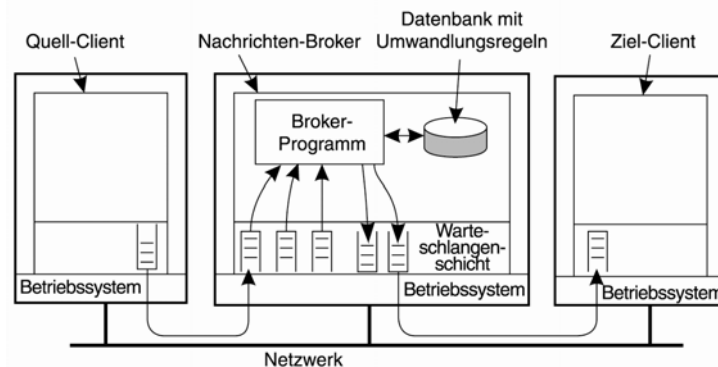


Abbildung: Nachrichten-Broker

Im folgenden Beispiel wird exemplarisch die Notwendigkeit von zentralen Konvertierungssystemen dargestellt.

Kommunikationsserver im Krankenhausbereich

In einem Krankenhaus, mit einer Vielzahl von unabhängigen Abteilungen und IT-Lösungen von sehr unterschiedlichen Firmen, übernimmt ein sogenannter Kommunikationsserver (z.B. von den Firmen e*Gate, Cloverleaf) die Aufgabe, Nachrichten in unterschiedliche Formate zu konvertieren. Die verwendeten Nachrichtenformate (HL7 (Health Level Seven)⁵¹, EDIFACT (Electronic Data Interchange For Administration, Commerce and Transport)⁵², DICOM (Digital Imaging and Communications in Medicine)⁵³, ...) sind zwar zum Teil an bestimmte Abteilungen (z.B. DICOM in der Radiologie) gebunden, das medizinische Datenformat HL7 wird jedoch i.d.R. für die einrichtungswerte Kommunikation eingesetzt. Außerdem haben die jeweiligen Softwarelösungen meist unterschiedliche Versionen von HL7, bzw. herstellerabhängige Dialekte verwendet. Aus diesem Grund ist der Zwischenschritt der Nachrichtentransformation unumgänglich. Dazu ist es notwendig die Nachricht im Kommunikationsserver zu parsen und zu analysieren und anhand der Umwandlungsregeln eine neue Nachricht im Zielformat zu erzeugen. Diese Möglichkeit ermöglicht einem Krankenhaus die IT-Lösungen von sehr unterschiedlichen Firmen mit sehr verschiedenen Schwerpunkten (Verwaltung, Radiologie, Apotheken, Labore, Endoskopie, Küche, . . .) zusammenzuführen und damit Medienbrüche zu vermeiden und eine entsprechende und notwendige Rationalisierung genutzt werden kann.

⁵¹ <http://www.hl7.de>.

⁵² <http://www.edifactory.de/index.html>.

⁵³ <http://medical.nema.org>.

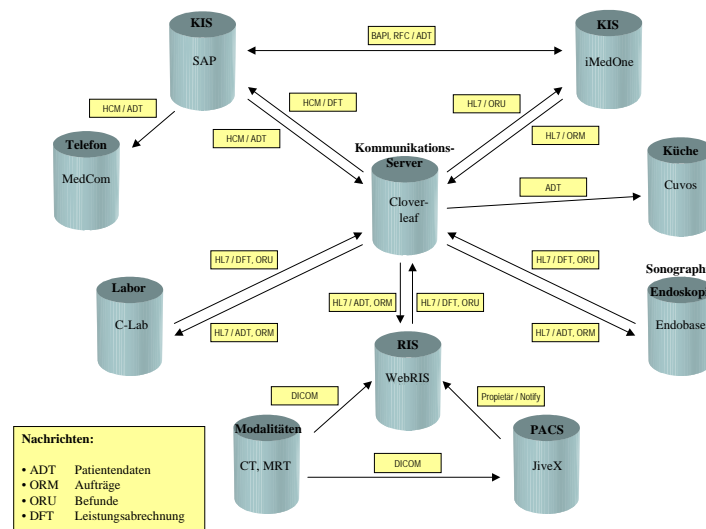


Abbildung: Beispiel der Schnittstellenvielfalt im Krankenhausbereich

Durch die große Anzahl der unterschiedlichen Systeme entstehen relativ komplexe Kommunikationsbeziehungen. Die Grafik zeigt beispielhaft eine Übersicht möglicher Systeme, zusammen mit den von ihnen ausgetauschten Nachrichten (Nachrichtenformat / Nachrichtentyp).

Das Beispiel eines Kommunikationservers im Krankenhaus zeigt, dass Firmen, die in sehr unterschiedlichen Bereichen/Branchen Softwareentwicklung betreiben, dennoch die Notwendigkeit haben, Informationen austauschen zu müssen, um ein Gesamtsystem betreiben zu können, um kostspielige Medienbrüche zu vermeiden und besondere Anforderungen wie ein zentrale Archivierung zu ermöglichen.

6.1.3 Diskussion über die Herstellung von Interoperabilität

Da nicht nur eine Firma den Softwaremarkt in allen Bereichen/Branchen beherrschen kann und soll ist die Notwendigkeit über Konvertierung von Repräsentationen und Formate, unterschiedlichen Anwendungen zusammenzubringen, unumgänglich. Aus diesem Grund spielt die Konvertierung für die Erzielung einer Interoperabilität eine enorm wichtige Rolle, um die unterschiedlichen Softwaremärkte nebeneinander bestehen zu lassen.

Hier sind ganz unterschiedliche Branchen betroffen wie z.B. im Krankenhausbereich die interne Logistik, Medizin, Controlling, usw. D.h. in den betreffenden Sektoren müssen die Firmen Spezial Know-how erworben haben, um ihre Software erstellen zu können. Das geht wohl verloren, wenn ihre Softwarelösungen im Konzert aller Anwendungen, die in einem Krankenhaus laufen, nicht mehr mitspielen können. Wenn die Software (Softwaremodule) Verfahren (Konvertierungsprogramme, Nachrichten-Broker) betreffen, die für die Herstellung von Interoperabilität notwendig sind, ist der Inhaber des Patentes in der Lage, den Wettbewerb erheblich zu beeinflussen, ohne selbst Wertschöpfung eingebracht zu haben.

6.2 Hardwareinterfaces

6.2.1 Beschreibung der Aufgabenstellung

Ein Hardwareinterface beschreibt, wie ein Softwaremodule die Funktionen einer Hardware nutzen kann. Damit die Software dies tun kann, muss die Schnittstelle gut dokumentierte werden.

Im folgenden Beispiel wird exemplarisch ein Beispiel aus der Praxis dargestellt:

Ein mögliches Szenario ist z.B., wenn es auf dem Markt Hardwarekarten mit entsprechenden Treibern für die Microsoftwelt gibt, aber keine Treiber für Linux zur Verfügung gestellt werden. Der Käufer möchte aber die Hardwarekarte unter Linux verwenden. Hier bleibt i.d.R. nur die Möglichkeit mit den Mechanismen des Reverse Engineering das Hardwareinterface zu analysieren und einen eigenen Treiber für Linux zu programmieren. Die Hardwarekarte kann z.B. eine Graphikkarte, TV-Karte oder eine Kommunikationskarte sein. Das Reverse Engineering funktioniert i.d.R. so, dass unter einem Microsoft-Betriebssystem der Treiber mit Hilfe eines Debuggers durchlaufen wird und man erkennen kann, wie man die einzelnen Register der Hardwarekarte bedienen muss, damit die Funktionen über eine Software genutzt werden können. Manchmal werden zusätzlich Hardwareanalytoren verwendet, um ein genaues Bild der Schnittstelle zu erlangen.

Wenn die Hardwareschnittstelle so analysiert wurde, kann z.B. ein Treiber für Linux realisiert werden.

Hinweis:

Da es heute schon eine Vielzahl an Treibern für Linux gibt, ist diese Vorgehensweise nur bei speziellen Hardwarekarten notwendig. Manche Firmen stellen auch eine Beschreibung des Hardwareinterface zur Verfügung. In der Praxis hat sich nur herausgestellt, dass diese Beschreibungen oft nicht ausreichen, um die Schnittstelle umfänglich zu bedienen.

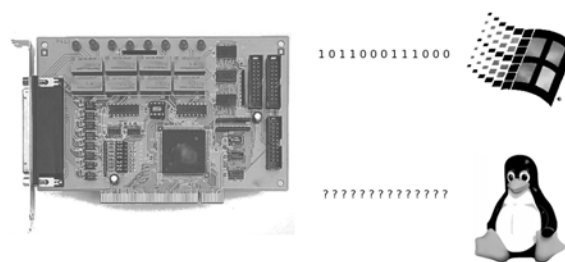


Abbildung: Kartewarekarte

Beispiele für solche Hardwarekarten, bei denen ein Reverse Engineering notwendig ist sind: Kommunikationskarten, Graphikkarten, TV-Karten⁵⁴⁵⁵, usw.

⁵⁴ <http://www.linux-club.de/viewtopic.php?t=12030>, siehe Beispiele dort.

⁵⁵ <http://www.linuxtv.org/download/dvb/>, siehe Beispiel dort.

6.2.2 Diskussion über die Herstellung von Interoperabilität

Hardware Investitionen können nur durch andere Softwaremodule wie z.B. Betriebssysteme weiter verwendet werden, wenn die Schnittstellen bekannt sind.

6.3 Software Reverse Engineering, Dekompilation von Software

6.3.1 Beschreibung der Aufgabenstellung

Software Reverse Engineering ist der umgekehrte Entwicklungsprozess von Software, der in einem Objektcode zur Verfügung steht. Ausgehend von einem existierenden Softwareprogramm im Objektcode, wird durch Analyse der Komponenten dessen Funktionsweise ermittelt. Diese Vorgehensweise ist nur notwendig, wenn man keinen Zugriff auf die Pläne, Unterlagen oder Quellen des Originals hat. Reverse Engineering funktioniert durch Dekompilation des Objektcodes. Dadurch kann die Ausführung des Computerprogramms z.B. auf der untersten Ebene der Maschinensprache (Quellcode) beobachtet werden. Erfahrene Praktiker sind dann in der Lage zu erkennen, wie das Computerprogramm arbeitet und wie die Schnittstelle realisiert ist.

Speziell bezogen auf Computer-Software wird darunter meistens einer der beiden folgenden Vorgänge verstanden:

- Die Rückgewinnung des Quellcodes oder einer vergleichbaren Beschreibung aus Binärcode, z. B. von einem ausführbaren Programm oder einer Programmbibliothek.
- Die Erschließung der Regeln eines Kommunikationsprotokolls aus der Beobachtung der Kommunikation, z.B. mit Hilfe eines Sniffers.

Im ersten Falle werden oft Decompiler eingesetzt, die den Quellcode eines Programms weitestgehend automatisch aus seinem Binärcode zurückgewinnen. Ist dies nicht durchgehend möglich, so kann auch der aus dem Binärcode des Programms direkt ermittelbare Maschinencode auch manuell analysiert werden, was allerdings eine merkliche Erschwernis bedeutet. Es kann i.d.R. nicht der komplette Programmquellcode ermittelt werden, da z. B. Kommentare und lokale Objektnamen nur selten im verfügbaren Binärcode enthalten sind. D.h. ein 100%iges Reverse Engineering ist nicht möglich. Meistens ist das Ergebnis eines Reverse Engineering jedoch trotzdem für den jeweiligen Zweck ausreichend und daher sinnvoll.

Viele Firmen untersagen das Reverse Engineering ihrer Produkte durch entsprechende **Lizenzbedingungen**. Die Analyse von Protokollen ist davon rechtlich nicht betroffen, weil dabei die Software selbst gar nicht Gegenstand der Untersuchung ist⁵⁶.

6.3.2 Diskussion über die Herstellung von Interoperabilität

Wie einige Beispiele (Festplattenverschlüsselung, Samba-Server, Hardwarekarten) aufzeigen, muss z.Z. Reverse Engineering betrieben werden, damit eine Interoperabilität erzielt werden kann. Da das Reverse Engineering nicht einfach ist, sehr viel Aufwand kostet und letztlich eine 100%tige Interoperabilität oft nicht erreicht werden kann, sind alle anderen Methoden (Standar-

⁵⁶ http://de.wikipedia.org/wiki/Reverse_engineering.

disierung und Normung, Offene Schnittstellen, usw.) sicherlich vorzuziehen, insbesondere eine gute Beschreibung der Schnittstellen und Nutzungsbeispiele durch den Hersteller, z.B. als SDK. Dennoch ist Reverse Engineering das einzige Mittel, welches unabhängig vom Hersteller zur einer Interoperabilität führt und auf das daher im IT-Markt nicht verzichtet werden kann.

6.4 Standardisierung und Normung

6.4.1 Beschreibung der Aufgabenstellung

Standardisierung und Normung sind Grundvoraussetzungen für netz- und grenzüberschreitende, weltweite Kommunikation.

Bei Standardisierung und Normung lassen sich vier Klassen von Gremien unterscheiden:

- Internationale Standardisierung (z.B. ISO, ITU, ...)
- Regionale Standardisierung (z.B. ETSI, DIN, ANSI, ...)
- Foren und Konsortien wie z.B. ATM-Forum, aber auch die Internet Engineering Task Force (IETF) passt in diese Kategorie
- Kooperationen zwischen Firmen („Initiativen“) bzw. „de facto“-Standards von Firmen

Bei den Begriffen für die Ergebnisdokumentation der Standardisierung gibt es eine verwirrende Vielfalt. „Standard“ ist ein nicht fest definierter Begriff. So benutzen sowohl anerkannte Standardisierungsgremien, als auch Konsortien und sogar Firmen intern diesen Begriff. Der Begriff **Standard** im Sinne der Standardisierung wird definiert als:

Dokument mit Festlegungen, das von einem Kreise von interessierten Fachleuten (Konsortium) erarbeitet wird. Es sind nicht immer alle Kreise einbezogen; ein Konsens ist nicht die Voraussetzung für eine Verabschiedung der Ergebnisse.

Anders sieht es bei „**Norm**“ aus. Es ist ein im Konsens aller Betroffenen erstelltes Dokument, das eine öffentliche Anhörung passiert hat und in das nationale bzw. internationale Normenwerk (z.B. DIN in D, ISO) übernommen wurde. Auch viele ETSI-Standards wurden in nationale Normen überführt. Daher sollte alles, was keinen offiziellen Status hat, „Spezifikation“ genannt werden. Das ATM-Forum z.B. benutzt korrekterweise diesen Begriff. Die Internet Engineering Task Force ist hier noch zurückhaltender mit der Terminologie und nennt seine Dokumente „Request for Comment“ (RFC), erhebt aber einige Schlüssel-RFCs zu „Standards“. Eine völlig andere Situation liegt bei ITU vor. Die Ergebnisdokumente werden „Empfehlungen“ (Recommendation) genannt. Sie sind im strengen Sinne nicht bindend. Manche Gremien erarbeiten zusätzlich „Technische Reports“. Dabei handelt es sich um informative und erläuternde, sog. nicht-normative Dokumente. Aber unabhängig davon, welcher Begriff letztlich verwendet wird lässt sich beobachten, dass sobald ein breiter Konsens unter allen Beteiligten besteht, ein entsprechendes Dokument einen Stellenwert erlangt, der von der Softwareindustrie nicht ignoriert werden kann. In diesem Zusammenhang bedeutet Konsens, dass mehrere Firmen Lösungen realisieren und damit einen Markt schaffen.

Die Bedeutung von Standards

Nach der Beschreibung von Herr Dr. Wende, Geschäftsführer des DIN (Deutsches Institut für Normung e. V.) "stellen Internationale Standards bzw. Normen eine Grundlage für Interoperabilität dar". Bezüglich der Patente beschreibt Dr. Wende "In Normen soll die Bezugnahme auf Patente vermieden werden. Ist das nicht vermeidbar, so muss sich der Patentinhaber verpflichten, die Nutzung seines Patenten allen Interessierten zu fairen und nicht diskriminierenden Bedingungen einzuräumen. Ist das nicht gewährleistet, so kann die Norm nicht verabschiedet werden."⁵⁷

Einige Kernaussagen der Gemeinschaftsstudie "Gesamtwirtschaftlicher Nutzen der Normung" der Technischen Universität Dresden und des Fraunhofer Instituts für Systemtechnik und Innovationsforschung sind⁵⁸:

- Über 15 Mrd. EUR jährlich beträgt der volkswirtschaftliche Nutzen der Normung.
- Wirtschaftswachstum wird durch Normen stärker beeinflusst als durch Patente und Lizenzen.
- Wissens- und Zeitvorteile erzielen Unternehmen, die sich an der Normungsarbeit beteiligen.
- Transaktionskosten werden gesenkt, wenn europäische und internationale Normen angewendet werden.
- Das Forschungsrisiko und die Entwicklungskosten werden für alle am Normungsprozess Beteiligten reduziert.

Im Folgenden werden die prinzipiellen Aussagen der "Intellectual Property Rights Policy" einiger Gremien dargestellt.

ETSI⁵⁹ / ISO⁶⁰ / IEC / ITU⁶¹

Die Gremien sind darauf angewiesen, dass die Firmen und sonstige Organisationen IP-Rights angeben und übernehmen keine Verantwortung für eine vollständig Recherche. Wenn ein Gremium Kenntnisnahme von wichtigen IPR bekommt, die einen Standard betreffen, werden die IPR-Besitzer aufgefordert, schriftlich eine nicht widerrufbare Zusage von "fair, reasonable and non-discriminatory terms and conditions" zu gewähren. Ist dies nicht gewährleistet, so wird der Standard i.d.R. entweder nicht verabschiedet, oder so überarbeitet, dass die Kollision mit dem „blockierenden“ Schutzrecht vermieden wird.

W3C⁶²

⁵⁷ Siehe auch DIN 820 Teil 1, Absatz 5.8 .

⁵⁸ <http://www.normung.din.de/>.

⁵⁹ <http://www.etsi.org/>.

⁶⁰ <http://www.iso.org/iso/en/ISOOnline.openerpage>.

⁶¹ <http://www.itu.int/home/index.html>.

⁶² <http://www.w3.org/>.

Hier müssen die Mitglieder Statements abgeben, in denen sie versichern, dass sie im Bereich der Standards keine Schutzrechte innehaben.

IETF⁶³

Die Internet Engineering Task Force (IETF) geht anders mit IPRs um. Bis auf eine Ausnahme im Bereich Sicherheitstechnologie, wird argumentiert, dass im Standardisierungsprozess notwendig ist, nachzuweisen, dass unterschiedliche Implementierung von unterschiedlichen Firmen erfolgreich miteinander laufen. Wenn dies der Fall ist wird angenommen, "that the terms are reasonable and to some degree non-discriminatory" gewährleistet ist. Hier wird die Verantwortung auf die Treiber eines Standards gelegt, was scheinbar bis jetzt gut funktioniert hat.

"Intellectual Property Rights in IETF Technology (RFC 2668) - Auszug

"4.1. No Determination of Reasonable and Non-discriminatory Terms

The IESG will not make any explicit determination that the assurance of reasonable and non-discriminatory terms or any other terms for the use of an Implementing Technology has been fulfilled in practice. It will instead apply the normal requirements for the advancement of Internet Standards. If the two unrelated implementations of the specification that are required to advance from Proposed Standard to Draft Standard have been produced by different organizations or individuals, or if the "significant implementation and successful operational experience" required to advance from Draft Standard to Standard has been achieved, the IESG will presume that the terms are reasonable and to some degree non-discriminatory. (See RFC 2026, Section 4.1.3.) Note that this also applies to the case where multiple implementers have concluded that no licensing is required. This presumption may be challenged at any time, including during the Last-Call period by sending email to the IESG⁶⁴.

8. Evaluating alternative technologies in IETF working groups

In general, IETF working groups prefer technologies with no known IPR claims or, for technologies with claims against them, an offer of royalty-free licensing. But IETF working groups have the discretion to adopt technology with a commitment of fair and non-discriminatory terms, or even with no licensing commitment, if they feel that this technology is superior enough to alternatives with fewer IPR claims or free licensing to outweigh the potential cost of the licenses.

Over the last few years the IETF has adopted stricter requirements for some security technologies. It has become common to have a mandatory-to-implement security technology in IETF technology specifications. This is to ensure that there will be at least one common security technology present in all implementations of such a specification that can be used in all cases.

This does not limit the specification from including other security technologies, the use of which could be negotiated between implementations. An IETF consensus has developed that no mandatory-to-implement security technology can be specified in an IETF specification unless it has no known IPR claims against it or a royalty-free license is available to implementers of the specification unless there is a very good reason to do so. This limitation does not extend to other security technologies in the same specification if they are not listed as mandatory-to-implement."

⁶³ <http://www.ietf.org/>.

⁶⁴ <http://www.ietf.org/rfc/rfc3668.txt?number=3668>.

Hinweis:

Der letzte Abschnitt unterstreicht die Anforderung, dass IT-Sicherheitstechnologie für die Interoperabilität eine besondere Rolle spielt.

6.4.2 Diskussion über die Herstellung von Interoperabilität

Standardisierung und Normung ist ein sehr geeignetes Mittel, um Konsens und Interoperabilität einfach zu erzielen. Aus diesem Grund sollten immer dann, wenn standardisierte Schnittstellen in einem Bereich vorhanden sind, diese auch verwendet werden.

Nach unserer Einschätzung kann die beschriebene positive Entwicklung von Interoperabilität durch politische Maßnahmen unterstützt und gefördert werden. Ein Ziel muss dabei sein, gerade deutsche Unternehmen von der Bedeutung und Wirksamkeit der Mitarbeit in nationalen und internationalen Standardisierungsorganisationen zu überzeugen.

6.5 Offen Schnittstellen

6.5.1 Beschreibung der Aufgabenstellung

Ein offene Schnittstelle bedeutet, dass alle notwendigen Informationen zur Erlangung einer Interoperabilität zu Verfügung gestellt werden.

OpenSource

Hier wird der gesamte Quellcode zur Verfügung gestellt, mit dessen Hilfe die Schnittstelle, wie Syntax und Semantik des In- und Output, ermittelt werden kann. Damit kann die Interoperabilität und damit auch die ordnungsgemäße Nutzung eines Softwaremoduls einfach gewährleistet werden⁶⁵.

6.5.2 Diskussion über die Herstellung von Interoperabilität

Eine offene Schnittstelle ist die einfachste und beste Möglichkeit, eine Interoperabilität zu gewährleisten. Aus diesem Grund sollten die eigentlichen Anwender von Firmen, die Software anbieten, die Offenlegung der Schnittstellen verlangen.

6.6 A rough consensus of a running code

Im RCF 1958 "Architectural Principles of the Internet" der Internet Engineering Task Force (IETF) wird beschrieben, auf welcher Grundlage die Entwicklung im Internet vorangetrieben wird. Besondere Aspekte sind:

- Fortunately, nobody owns the Internet, there is no centralized control, and nobody can turn it off.

⁶⁵ Vgl. GSP der Fa. Microsoft, www.heise.de/newsticker/meldung/51244.

- Its evolution depends on rough consensus about technical proposals, and on running code.
- Engineering feed-back from real implementations is more important than any architectural principles⁶⁶.

Dies bedeutet in der Praxis, dass mehrere Unternehmen eine Realisierung eines Standards (RCFs) nur dann vornehmen, wenn sie von der Notwendigkeit überzeugt sind. Außerdem ist es im Standardisierungsprozess notwendig nachzuweisen, dass unterschiedliche Implementierung von unterschiedlichen Firmen miteinander laufen. Aus diesem Grund arbeiten unterschiedliche Firmen sehr eng und offen miteinander, um die Interoperabilität zu gewährleisten.

Die IETF geht davon aus, dass wenn es mehrere Realisierungen von verschiedenen Firmen/Hochschulen gibt, das Interoperabilitätsproblem im Prinzip gelöst ist⁶⁷.

6.7 Testsysteme für die Interoperabilität

6.7.1 Beschreibung der Aufgabenstellung

Ein Testsystem ist eine Menge von Prozessen, Verfahren, Werkzeugen und Daten, mit denen - idealerweise vollautomatisch - die Funktionstüchtigkeit einer Schnittstelle überprüft werden kann. Teilweise werden Testsysteme von Prüflaboren wie z.B. den TÜVs zur Verfügung gestellt. Diese führen die Schnittstellentests durch und geben entsprechende Protokolle, die den Firmen helfen zu erkennen, an welchen Stellen sie möglicherweise noch nicht interoperable sind. Falls alle Bedingungen erfüllt sind, werden evtl. auch Zertifikate ausgestellt, mit denen die Firmen die Interoperabilität nach außen darstellen können.

Beispiel: ISIS-MTT

Die Bemühungen um Interoperabilität von PKI-Systemen und deren Anwendungen werden durch die Vielfalt der Standards und ihrer Interpretationsmöglichkeiten erschwert. Um eine gemeinsame Grundlage zu schaffen, haben die Vereinigungen TeleTrust e.V. und T7 e.V. mit Unterstützung durch das BMWA die Spezifikation **ISIS-MTT** ins Leben gerufen. Deren übergeordnetes Ziel ist eine schnelle flächendeckende Einführung PKI-gestützter Sicherheitstechnologien.

Folgende Kriterien werden hierbei zugrunde gelegt:

- Investitionssicherheit für Anwender und Anbieter durch Kompatibilität zu internationalen Standards und bereits verwendeten Zertifikatsformaten, durch langfristige Stabilität der Spezifikation und problemlose Interoperabilität von Produkten verschiedener Hersteller
- Anpassungsfähigkeit des Sicherheitsniveaus an die unterschiedlichen Anwendungsfelder.
- Berücksichtigung aller für den elektronischen Geschäftsverkehr relevanten Formvorschriften (z.B. des Privat- und Verwaltungsrechts).

⁶⁶ Siehe RFC 1958; www.ietf.org.

⁶⁷ <http://www.ietf.org/rfc/rfc3668.txt?number=3668>, Kap. 8.

- Die ISIS-MTT-Spezifikation basiert auf internationalen Standards (S/MIME, PKIX, PKCS, X.509, ETSI, CEN ESI etc.) und integriert die aus der bisherigen Anwendung gewonnenen Erfahrungen. Sie besteht aus einer Kernspezifikation, die die Grundanforderungen aller Anwendungskategorien abdeckt, und Erweiterungen für die verschiedenen Anwendungen (z.B. elektronische Signatur, E-Mail-Verschlüsselung). Darin werden ausschließlich Festlegungen getroffen, die in den bestehenden Standards nicht hinreichend eindeutig geregelt sind⁶⁸.

6.7.2 Diskussion über die Herstellung von Interoperabilität

Gerade bei komplexen Schnittstellen kann mit Hilfe von Testsystemen die Interoperabilität stark gefördert werden. Voraussetzung ist allerdings, dass die Beschreibungen der Schnittstellen zur Verfügung stehen.

7 Zusammenfassung: Interoperabilität

Im Bereich der Software ist es gängige Praxis und auch eine wichtige Notwendigkeit, dass unterschiedliche Softwaresysteme wie z.B. Betriebssysteme, Anwendungen, Tools (Libraries oder sonstige Dienststroutinen, wie Middleware, usw.) und IT-Services, miteinander verwoben werden können, um so gemeinsam Aufgaben und Funktionen in einem Gesamtsystem mit teilweise unterschiedlichen Aufgabenstellungen und Branchenwissen lösen zu können.

Mit der Hilfe von Patenten können sehr große Unternehmen (insbesondere aus den USA) die Interoperabilität ihrer Software zu den anderen Herstellern beschränken, um ihre eigene Marktposition weiter zu verbessern.

Aus diesem Grund ist die Interoperabilität eine wesentliche Voraussetzung für die Funktionsfähigkeit des Wettbewerbs und von daher von zentraler wirtschaftlicher Bedeutung.

Aus diesem Grund muss eine angemessene Interoperabilitätsklausel in der Richtlinie berücksichtigt werden.

Vorschlag für die Richtlinie aus informationstechnologischer Sicht:

Für die Schnittstellen computerimplementierter Erfindungen sind von den Patentinhabern so gute Beschreibungen zur Verfügung zu stellen, dass es jedem Benutzer der Schnittstellen mit relativ geringem Aufwand ermöglicht wird, Interoperabilität zu computerimplementierten Erfindung herstellen zu können.

Falls diese Beschreibung nicht ausreicht, muss mit Mitteln des Reverse Engineering zu Zwecken der Interoperabilität die computerimplementierten Erfindungen ohne Zustimmung des Herstellers analysiert werden können.

⁶⁸ Blumberg/Pohlmann, S. 238.

8 Rechtsrahmen und Wirkung von Schutzrechten im Rahmen der Herstellung von Interoperabilität

Die wirtschaftliche Bedeutung sowie die Struktur des Softwaremarktes in Deutschland wurde ausführlich im Kapitel „**Softwaremarkt**“ dargestellt. Berücksichtigt werden muss, dass dieses Marktsegment keine rein nationale Prägung erfahren hat, sondern vielfältigen internationalen Einflüssen ausgesetzt ist. Aufgrund der Erwartung auf hohem Niveau sich fortschreibender Wachstumspotentiale, insbesondere mit Blick auf die Bedeutung des elektronischen Geschäftsverkehrs, wird dem Softwaremarkt eine Schlüsselrolle hinsichtlich der künftigen wirtschaftlichen Entwicklung in Deutschland und Europa beigemessen. In besonderer Weise gilt dies für die aus dieser Entwicklung zu erwartenden positiven Rückwirkungen auf den Arbeitsmarkt.

Da die Herstellung von Softwarelösungen im Regelfall mit hohen Entwicklungskosten verbunden ist, kommt der Frage des rechtlichen Schutzes von Software zentrale Bedeutung zu, zumal sich abzeichnet, dass die Softwareindustrie mittlerweile einen Reifegrad erreicht hat, der die Erzielung von Softwareverbesserungen zunehmend erschwert und damit kostenintensiver gestaltet. An Brisanz hat die Thematik zudem vor allem durch die Tatsache gewonnen, dass in wachsendem Umfang Open Source Software zum Einsatz gelangt und die Hersteller proprietärer Softwareprodukte sich in Folge dessen einem erhöhten Wettbewerbsdruck ausgesetzt sehen⁶⁹. Ferner besteht durch die Entstehung von sog. „Quasi-Monopolen“ die latente Gefahr, dass Hersteller eine wettbewerbsfeindliche Geschäftspolitik durch die Nichtoffenlegung der für die Herstellung von Interoperabilität erforderlichen Informationen betreiben. Beispielhaft ist das von der EU-Kommission gegen die Fa. Microsoft nach Art. 82 EGV⁷⁰ eingeleitete Verfahren. Microsoft wurde verpflichtet die für die Verbindung von Arbeitsgruppenserver-Betriebssystemen notwendigen Informationen binnen 120 Tagen offen zu legen. Auch in den U.S.A. war Microsoft derartigen Vorwürfen im Rahmen von „Anti-Trust-Verfahren“ ausgesetzt, da sie den Vertrieb von Browser-Software, zur Marktdurchsetzung des eigenen Produkts – Internet Explorer, versuchte durchzusetzen⁷¹.

Vor diesem Hintergrund soll im Folgenden untersucht werden, inwieweit die im technischen Teil des Gutachtens unter dem Blickwinkel der Herstellung von Interoperabilität vorgestellten technischen Lösungen dem Patentschutz zugänglich sind. Nachfolgend wird der Frage nachgegangen, ob bereits nach geltendem Recht die Herstellung von Interoperabilität gewährleistet ist. Auf der Grundlage der gewonnenen Ergebnisse soll eine Bewertung des Vorschlag für eine Richtlinie des Europäischen Parlaments und des Rates über die Patentierbarkeit computerimplementierter Erfindungen erfolgen⁷².

8.1 Softwaresysteme und Schnittstellen

Wie unter 4.3 dargestellt setzen sich Softwaresysteme aus miteinander verknüpften Softwaremodulen zusammen. Die Module unterschiedlicher Hersteller können (im Einzelnen unter 4.4

⁶⁹ Dies zeigen u.a. die jüngsten Diskussionen um die Frage, ob Öffentliche Auftraggeber (hier die Stadt München) Open Source Lösungen im Wege einer Freihändigen Vergabe beschaffen können.

⁷⁰ Entscheidung der Europäischen Kommission vom 24. März 2004 – Fall COMP/C-3/37.792 Microsoft.

⁷¹ *US v. Microsoft Corp.*, 253 F3d 34, 63 (D.C. Cir. 2001); *Mestmäcker/Schweitzer*, Europäisches Wettbewerbsrecht, 2. Aufl. § 28 Rdnr. 56f.

⁷² Vorschlag für eine Richtlinie des Europäischen Parlaments und des Rates über die Patentierbarkeit computerimplementierter Erfindungen, KOM (2002) 92 endg., S. 2.

dargestellt) nur dann miteinander verbunden werden, wenn diese untereinander die entsprechenden Informationen austauschen können (Input und Output).

Hierbei handelt es sich um:

- Betriebssysteme
- Kommunikationssysteme
- Verteilte Systeme
- Anwendungssysteme
- Trusted Computing, Nexus
- Kryptoalgorithmen, Sicherheitsprotokolle und Sicherheitsinfrastrukturen
- Konvertierungsprogramm
- Embedded Systems

Wie die Abbildung „Beschreibung der Funktion einer Schnittstelle“ verdeutlicht, ist die Schnittstelle kein Bestandteil der Hardware. Der Austausch von Input und Outputinformationen erfolgt somit ausschließlich zwischen Softwaremodulen.

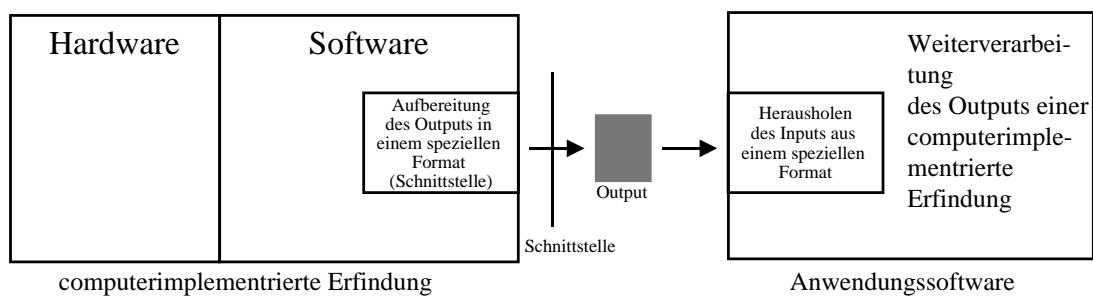


Abb.: Beschreibung der Funktion einer Schnittstelle

Die Diskussion um die Patentfähigkeit von computerimplementierten Erfindungen soll in diesem Gutachten nicht aufgegriffen werden. Zur Verdeutlichung der Rahmenbedingungen, unter denen die Beurteilung der nachfolgend zu untersuchenden Systeme zu erfolgen hat, werden ausschließlich beschreibend, die derzeit in der Patentpraxis an computerimplementierte Erfindungen anzulegenden Grundsätze dargestellt.

8.2 Schutzrechte - Patent- und Urheberrechtsschutz

Nach derzeitiger Rechtslage kann der Schutz von Computerprogrammen vor unberechtigter Nutzung auf zwei Wegen erfolgen. Einerseits auf der Grundlage des Urheberrechtsschutzes,

der derzeit noch vorherrschenden Form, bzw. des Patentrechts⁷³, wenn der Gegenstand der Patentanmeldung eine computerimplementierte Erfindung darstellt. Einen begrenzten Know-how-Schutz bietet ferner das Wettbewerbsrecht. Auslöser der zunehmenden Tendenz, computerimplementierte Erfindungen patentrechtlich schützen zu lassen ist eine Konsequenz aus der Tatsache der Verkürzung der Zeitspanne, in der Wettbewerber eines Softwareentwicklers in der Lage sind, eine Softwareinnovation zu imitieren⁷⁴. Die patentrechtliche Behandlung solcher Plagiate ist nicht Gegenstand dieser Untersuchung, Ziel ist vielmehr die Erörterung der Frage, ob der Inhaber einer computerimplementierten Erfindung durch das Patent, erfindungsferne Teilmärkte der Softwarebranche zu „monopolisieren“ vermag.

8.2.1 Voraussetzungen des Patentschutzes

Hinsichtlich der nach deutschem Patentrecht bestehenden gesetzlichen Grundlagen bezüglich der Patentfähigkeit von Computerprogrammen kann im Grundsatz auf die Analyse der Rechtslage in Deutschland auf den Forschungsbericht „Mikro- und makroökonomischen Implikationen der Patentierbarkeit von Softwareinnovationen“ verwiesen werden⁷⁵. Im Folgenden werden, ohne Anspruch auf Vollständigkeit, Merkmale, die im Rahmen der Feststellung der Patentfähigkeit computerimplementierter Erfindungen von besonderer Bedeutung sind, einer kurzen Betrachtung unterzogen.

8.2.1.1 Technizität

Im Rahmen der Entscheidung über die Patentfähigkeit kommt dem **Erfindungsbegriff** zentrale Bedeutung zu. Der Begriff der Technik fungiert nach der Rechtsprechung des BGH als Abgrenzungskriterium zwischen patentfähigen und anderen Leistungen, für die entweder kein Patentschutz vorgesehen ist oder sie nicht entsprechend geeignet sind.

Sind nicht-technische Merkmale in der Erfindung enthalten, stellt sich gerade im Zusammenhang mit computerimplementierten Erfindungen die Frage, ob die Implementierung eines neuen aber nicht-technischen Merkmals in einen bekannten technischen Gegenstand in der Gesamtbetrachtung zu einem insgesamt neuen technischen Gegenstand führt. Bejaht man diese Frage, können nicht-technische Merkmale einer Erfindung z.B. deren Neuheit begründen⁷⁶. Teilweise wird in Fortschreibung dieses Denkansatzes die Schlussfolgerung gezogen, dass entsprechend auch „Geschäftsmethoden, die als solche wie auch Computerprogramme vom Pa-

⁷³ In erster Linie verlassen sich Softwarehersteller noch immer auf den Urheberrechtsschutz. Ausweis für die gleichwohl wachsende Bedeutung von Patenten, ist die steigende Zahl von Patentanmeldungen in diesem Bereich. Im Jahr 2003 wurden beim DPMA 64.518 Anmeldungen eingereicht, was einen Zuwachs gegenüber dem Vorjahr von 1,7% bedeutet; vgl. DPMA Geschäftsbericht 2003, S. 6. Ferner hat sich die Zahl ausländischer Anmeldungen nach dem Patentrechtsabkommen (Patent Cooperation Treaty – PCT), BGBl II 1970, 649, 664 i.d.F. vom 2. Oktober 2001 BGBl II 2002, 728; die Anzahl der Mitgliedsstaaten beläuft sich auf 123 (Stand 1. Januar 2004). An der Spitze stehen Anmelder aus den U.S.A. mit 21,9% = 31.624 und aus Japan mit 13,8% = 18.373 Anmeldungen (DPMA Geschäftsbericht 2003, S. 8f.). Da die Anmeldungen den jeweiligen IPC-Klassen (Zum System der „Internationalen Patentklassifikation“ s. *Kraßer*, Patentrecht, S. 82f.) zugeordnet werden, liegt kein statistisches Material darüber vor, in welcher Höhe Anmeldungen dem Gebiet computerimplementierter Erfindungen zugeordnet werden können. Der Grund hierfür liegt in der Tatsache begründet, dass das IPC-Klassifikationssystem noch keine Merkmale enthält, die eine entsprechende Zuordnung möglich machen würde.

⁷⁴ *Weyand/Haase*, GRUR 2004, 198 (199) m.w.Nw.

⁷⁵ Mikro- und makroökonomischen Implikationen der Patentierbarkeit von Softwareinnovationen – Endbericht, S. 125ff. (Stand September 2001).

⁷⁶ *Tauchert*, Zum Begriff der „technischen“ Erfindung, JurPC Web-Dok. 28/2002, Abs. 345ff., 59.

tentschutz ausgeschlossen sind, im Fahrwasser einer in eine bestehende Erfindung eingebetteten computerimplementierten Erfindung, gleichfalls patentiert werden können⁷⁷. Damit ist der Weg eingeschlagen, die im PatG und EPÜ enthaltenen Ausschlussstatbestände ins Leere laufen zu lassen. Erforderlich ist nur die Verbindung mit einer industriell herstellbaren und gewerblich einsetzbaren Vorrichtung, was unschwer möglich sein soll⁷⁸.

„Technizität ist danach gegeben, wenn planmäßiges Handeln unter Einsatz beherrschbarer Naturkräfte zur Erreichung eines kausal übersehbaren Erfolges, der ohne Zwischenschaltung menschlicher Verstandestätigkeit die unmittelbare Folge beherrschbarer Naturkräfte ist“⁷⁹.

Dieser Technikbegriff wurde durch die Entscheidungen „Logikverifikation“, „Sprachanalyseeinrichtung“ und „Suche fehlerhafter Zeichenketten“ mit Blick auf den technischen Charakter von Computerprogrammen sowie auf Art. 27 Abs. 1 Satz 1 TRIPS-Übereinkommen durch den BGH verändert. Da es üblich geworden sei, Computer in praktisch allen Bereichen moderner Technik einzusetzen, müsse der modifizierte Technikbegriff für alle technischen Gebiete, deren technologische Entwicklung den Einsatz von entsprechenden Computerprogrammen erfordere, gelten. Nunmehr ist bereits die Umsetzung einer auf technischen Überlegungen beruhenden Erkenntnis in ein Programm für Datenverarbeitungsanlagen ausreichend, um die Technizität eines computerimplementierten Verfahrens zu begründen⁸⁰. Infolge dieser Entwicklung wird der Rückschluss gezogen, dass der Technikbegriff des Patentrechts nicht statisch ist, sondern Modifikationen infolge neuer technischer Entwicklungen zugänglich sein muss, um auf diesem Wege einen effektiven Patentschutz zu ermöglichen⁸¹.

Aus dem Begriff der Technizität ergibt sich somit unmittelbar, dass ein physikalischer Effekt Hauptzweck der Erfindung sein muss. Da Software notwendig auf einem physikalischen Datenträger gespeichert und durch eine Hardware ausgeführt werden muss, kann nach Sinn und Zweck dieser Regelungssystematik, die Verkörperung auf einem Datenträger jedoch nicht schon für sich genommen zur Patentfähigkeit einer computerimplementierten Erfindung führen. Entscheidend ist der sachliche Gehalt einer Erfindung. Er bildet die Grundlage für die Entscheidung darüber, ob eine Lehre technischer Natur ist, d.h. die technischen Elemente der Problemlösung im Vordergrund stehen⁸². In der Konsequenz folgt daraus, dass eine programmbezogene Lehre dann technisch ist, wenn sie die Funktionsfähigkeit der Datenverarbeitungsanlage als solche betrifft und so das Zusammenwirken ihrer Elemente ermöglicht⁸³. Reine Anwendungsprogramme vermögen diese Voraussetzung nicht zu erfüllen⁸⁴. Eine zweite Einschränkung liegt in der Voraussetzung, dass das technische Handeln ohne Zwischenschaltung menschlicher Verstandestätigkeit erfolgen muss, d.h. patentfähig sind allein automatisch arbeitende Systeme. In der Konsequenz daraus ergibt sich, dass an sich interaktive Programme (sog. „Interaktiver Dialog“) diese Voraussetzung nicht erfüllen können⁸⁵.

⁷⁷ Tauchert, Zum Begriff der „technischen“ Erfindung, JurPC Web-Dok. 28/2002, Abs. 345ff., 50.

⁷⁸ Anders, GRUR 2004, 461.

⁷⁹ BGH GRUR, 1980, 849 – Antilockiersystem.

⁸⁰ BPatGE 45, 103 = CR 2003, 18f. – Computerimplementiertes Verfahren zur Herstellung eines Kabelbaums.

⁸¹ BPatG, Jahresbericht 2002, S. 10.

⁸² BGH GRUR 1992, 36ff. – Chinesische Schriftzeichen.

⁸³ BGH GRUR 1992, 33 – Seitenpuffer.

⁸⁴ BGHZ 67, 22 – Dispositionsprogramm; BGH GRUR 1977, 657 – Straken; BGH GRUR 1978, 102 – Prüfverfahren; BGH GRUR 1978, 420 – Fehlerortung; BGH GRUR 1992, 36ff. – Chinesische Schriftzeichen.

⁸⁵ BPatGE 42, 208 – Demontageschwierigkeitsbewertung; BPatG, Beschluss v. 20. Mai 2003 (17 W (pat) 44/02), Mitt. 2003, 555 – Bedienhandlung – Rechtsbeschwerde zugelassen und eingelegt aber im Zeitpunkt der Erstellung des Gutachtens noch nicht entschieden (X ZB 33/03).

Bei der Feststellung der Patentfähigkeit orientiert sich die Rechtsprechung mithin am Technikbegriff und weniger an der Ausschlussliste des § 1 Abs. 2 PatG, Art. 52 Abs. 2 und 3 EPÜ, wie etwa die Entscheidungen „chinesische Schriftzeichen“, „Logikverfahren“ und „Suche fehlerhafter Zeichenketten“ aufzeigen⁸⁶.

Mittlerweile wird das Merkmal der Technizität aber auch kritisch beurteilt. In seinem Jahresbericht für das Jahr 2003 zitiert das BPatG *Keukenschrijver*, Richter am BGH im X. Senat, der mit Blick auf das Prüfungsmerkmal der Technizität feststellt:

„ Nach den Entscheidungen „Logikverifikation“, „Sprachanalyseeinrichtung“ und „Suche fehlerhafter Zeichenketten“ erweist sich die Technizität zunehmend als leicht zu nehmende Hürde und im Regelfall als nicht geeignet, Gegenstände vom Patentschutz auszuschließen“⁸⁷.

In diesem Zusammenhang stellt das BPatG ferner fest, dass es in seinen Entscheidungen zunehmend weniger häufig mit „Programmen für die Datenverarbeitung als solche“, sondern verstärkt mit Fragen der Technizität auf den Gebieten der Pläne, Regeln und Verfahren für gedanklich oder geschäftliche Tätigkeiten befasst wird⁸⁸.

Danach kann eine Unterscheidung in drei Grundtypen vorgenommen werden:

Technische Programme	Nicht-technische Programme mit technischem Charakter	Nicht-technische Programme mit nicht-technischem Charakter
Software, die für den Betrieb von Hardware unerlässlich ist und deren Nutzung erst ermöglicht	<p>Nicht-technische Programme haben nach der Rechtsprechung des BGH technischen Charakter und sind damit patentfähig, wenn</p> <ul style="list-style-type: none"> das Programm einen neuen, erfindnerischen Aufbau einer Datenverarbeitungsanlage lehrt oder ihm eine Anweisung zu entnehmen ist, die Anlage auf neue, bisher nicht übliche und auch nicht nahe liegende Weise zu benutzen (neue, erfindnerische Betriebsweise). 	<p>Es handelt sich um die sog. Anwenderprogramme. Sie werden von BGH nicht als patentfähig erachtet, da sie keine Teile der Hardware steuern, d.h. die Programme enthalten keine Anweisungen, die die Funktionsweise einer Datenverarbeitungsanlage unmittelbar betreffen, sondern die Anlage, wenn überhaupt, allenfalls über das Betriebssystem ansprechen.</p>

8.2.1.2 Bedeutung des Begriffs der Neuheit

Patente werden nur für Erfindungen erteilt, die neu sind (§§ 1 Abs. 1, 3 Abs. 1 Satz 1 PatG; Art. 54 Abs. 1 EPÜ), d.h. dass sie im Zeitpunkt der Anmeldung nicht zum Stand der Technik gehören. Dies gilt zunächst auch uneingeschränkt für computerimplementierte Erfindungen. Entscheidend ist dabei, dass die Erfindung eine Problemlösung aufzeigt, die sich vom Stand der bekannten Technik abhebt und einen neuen, bisher nicht bekannten Lösungsweg aufzeigt⁸⁹. Geprüft wird die Neuheit einer Erfindung auf der Grundlage eines Einzelvergleichs, wobei jeder zeitlich und inhaltlich möglicherweise neuheitsschädliche Sachverhalt für sich genommen auf die Frage hin zu prüfen ist, ob das Wissen, das durch ihn der Öffentlichkeit zugänglich gemacht wurde, die angemeldete Erfindung vorweggenommen hat⁹⁰. Ergebnis des Vergleiches mit dem

⁸⁶ *Tauchert*, Zum Begriff der „technischen“ Erfindung, JurPC Web-Dok. 28/2002, Abs. 49.

⁸⁷ Festschrift für Reimar König, S. 255 (262).

⁸⁸ BPatG, Jahresbericht 2003, S. 9

⁸⁹ *Ensthaler*, Gewerblicher Rechtsschutz und Urheberrecht, S. 114.

⁹⁰ *Kraßer*, Patentrecht, S. 284 n.w.N.

Stand der Technik ist, welches Erzeugnis, Stoff oder Verfahren zur Herstellung von Erzeugnissen als neu bezeichnet werden kann.

Für die Diskussion der Neuheit computerimplementierter Erfindungen kann festgestellt werden, dass Computerprogramme in Verknüpfung mit einer bekannten Vorrichtung Patentschutz erlangen können. Es handelt sich nicht um eine neue patentrechtliche Erkenntnis, dass ein Erzeugnis auch dann neu sein kann, wenn es bekannte Bestandteile enthält, die nur geringfügig verändert in einer bisher nicht bekannten Verknüpfung verwendet werden⁹¹. Nicht ausreichend ist dagegen, wenn eine bekannte Vorrichtung unverändert für einen neuen Zweck angewendet wird. Dem Patentschutz wäre in diesem Fall nur die neue Anwendung zugänglich⁹².

In der gerichtlichen Praxis stellt der Begriff der Neuheit nach Auffassung des BPatG regelmäßig kein Problem dar⁹³. Ob diese Einschätzung auf der Tatsache beruht, dass bisher nicht in nennenswertem Umfang die Frage nach der Feststellung des neuheitsschädlichen Standes der Technik gestellt wurde, kann im Rahmen dieses Kurzgutachtens nicht belastbar beantwortet werden.

Damit hängt die Patentfähigkeit computerimplementierter Erfindungen von dem Umfang des veröffentlichten Standes der Technik ab.

8.2.1.3 Das Merkmal der erfinderischen Tätigkeit

Was unter „erfinderischer Tätigkeit“ zu verstehen ist, wird durch § 4 PatG, Art. 56 S. 1 EPÜ definiert. Danach ist erfinderisch, was sich für den **Fachmann** nicht in nahe liegender Weise aus dem Stand der Technik ergibt. Zur Beurteilung können auch die tägliche Lebenserfahrung, allgemeines Fachwissen sowie elementares Grundwissen herangezogen werden. Angewendet wurden diese Grundlagenkenntnis vom BPatG im Rahmen der Beurteilung einer Patentanmeldung, die ein Kraftfahrzeug-Steuergerät betraf, welches Informationen über Umprogrammierungen eines Speichers betraf. Die Patentfähigkeit wurde unter Heranziehung allgemeinen Fachwissen und der täglichen Lebenserfahrung verneint. Begründet wurde die Entscheidung damit, dass der Fachmann, ein Programmierer, der die kraftfahrzeugspezifischen Vorgänge kenne, über umfassende Kenntnisse in der Datenverarbeitung verfüge. Deshalb seien ihm übliche Maßnahmen, wie das Protokollieren von Zugriffen auf sensible Daten, vertraut und im Übrigen zeige die tägliche Lebenserfahrung, dass Daten über Versuche von Speicherzugriffen gespeichert würden (Umgang mit Geldautomat)⁹⁴. Gerade in diesem zentralen Bereich der Feststellung von Patentfähigkeit herrscht ein beträchtliches Maß an Unsicherheit, da jeweils Einzelfallbezogene Entscheidungen zu treffen sind. Vorstellbar ist daher, dass eine Reihe von patentierten computerimplementierten Erfindungen, würde man die Grundsätze aus der Entscheidung „Kraftfahrzeug-Steuergerät“ anwenden, keinen Patentschutz erlangt hätten. So etwa das „Adobe-Karteikartenreiter-Patent“⁹⁵. Papp-Karteikarten mit Reitern sind seit Jahrzehnten als Ordnungssystem bekannt. Es könnte an dieser Stelle daher, unter Heranziehung allgemeinen Fachwissens sowie der täglichen Lebenserfahrung, beurteilt werden, dass das Unterteilen des Bildschirms in einen großen und einen kleinen Bereich, als eine für den Fachmann nahe liegende und vertraute Lösung ist. Es verbleibt somit in diesem Bereich an einem hohen Maß von Rechtsunsicherheit.

⁹¹ BGH GRUR 1979, 149 (150f.) – Schießbolzen.

⁹² Kraßer, Patentrecht, S. 291.

⁹³ BPatG, Jahresbericht 2003, S. 14.

⁹⁴ BPatG, Beschl. v. 22. November 2002, 17 W (pat) 59/00 – Kraftfahrzeug - Steuergerät; umfänglich zu allgemeinen Beurteilungskriterien im Rahmen der Feststellung der erfinderischen Tätigkeit, Kraßer, Patentrecht, S. 306ff.

⁹⁵ EPA Patent EP0689133.

Gleiches gilt für die Feststellung der Erfindungshöhe, der im Rahmen von Nichtigkeitsklagen nach allgemeiner Erfahrung die größte Bedeutung zukommt. Maßstab ist die Feststellung, dass der Zweck des Patentschutzes darauf gerichtet ist, den technischen Fortschritt in besonderer Weise zu fördern, soweit er nicht schon von der laufenden Anwendung des Standes der Technik zu erwarten ist. Patentschutz soll somit für Entwicklungen gewährt werden, die sich nicht aus „normaler“ Entwicklung des Standes der Technik ergeben⁹⁶. Sie ist also als ein qualitatives Erfordernis anzusehen. Da es sich jeweils um Einzelfallentscheidungen handelt, kann ihnen nur schwerlich weitergehende Bedeutung beigemessen werden, meist jedoch nur dann, wenn vergleichbare Fallgestaltungen vorliegen⁹⁷, wie dies das o.g. Beispiel zeigt.

8.2.2 Patentschutz für „Computerimplementierte Erfindungen“ nach PatG

Nach deutschem Patentrecht sind Computerprogramme „**als solche**“ nach § 1 Abs. 2 PatG dem Grundsatz nach nicht schutzfähig. Durch § 1 Abs. 3 PatG wird diese Entscheidung jedoch relativiert, indem der Ausschluss von der Patentfähigkeit nur für Software „als solche“ greifen soll.

Die Gründe für diesen Regelungsansatz werden in dem Forschungsbericht „Mikro- und makroökonomischen Implikationen der Patentierbarkeit von Softwareinnovationen“ näher erläutert⁹⁸. Wesentlich und daher im Rahmen der weiteren Diskussion ist die Tatsache besonders zu berücksichtigen, dass im Zeitpunkt der Einfügung des Patentierungsverbotes für Computerprogramme noch weitgehend ungeklärt war, unter welchen Voraussetzungen von der Patentfähigkeit eines Computerprogramms, respektive computerimplementierter Erfindungen, ausgegangen werden konnte. Schon damals hat der BGH in seiner Rechtsprechung die Frage der Patentfähigkeit vom Computerprogrammen offen gehalten und bereits in seinen ersten Entscheidungen zu diesem Fragenkomplex diese sogar grundsätzlich für möglich gehalten⁹⁹. Kurz und verkürzt gefasst kann man feststellen, dass die Software Bestandteil eines übergeordneten Systems sein muss. Dabei kommt der Software die Funktion zu das Wissen über die in der Erfindung enthaltene technische Leistung zur Verfügung zu stellen.

Steuert danach ein Computerprogramm eine Hardware, die für sich genommen zweifelsfrei als technisches Gerät zu qualifizieren ist, kommt der computerimplementierten Erfindung **technischer Charakter** zu¹⁰⁰. Nicht technisch sei dagegen lediglich die abstrakte, dem Programm zugrunde liegende Idee, die ebenso wie **Geschäftsmethoden** dem Patentschutz nach deutschem Patentrecht unstreitig nicht zugänglich sind. Teilweise wird in der Literatur die Auffassung vertreten, dass mit der „Ausweitung“ des Patentschutzes für computerimplementierte Erfindungen durch die Entscheidungspraxis von DPMA und EPA, die durch die Rechtsprechung auch des BGH gestützt würde, auch nach deutschem Patentrecht von einer „faktischen Patentfähigkeit“ von Computerprogrammen ausgegangen werden müsse¹⁰¹. In diesen Zusammenhang kann, ausgelöst durch das BGH-Urteil – „Suche fehlerhafter Zeichenketten“, auch die Diskussion zu der Frage gestellt werden, inwieweit nichttechnische Merkmale einer Erfindung auf

⁹⁶ Vgl. auch *Kraßer*, Patentrecht, S. 299.

⁹⁷ DPMA, Geschäftsbericht 2003, S. 14f.

⁹⁸ Mikro- und makroökonomischen Implikationen der Patentierbarkeit von Softwareinnovationen – Endbericht, S. 128 ff.

⁹⁹ *BGHZ* 67, 22ff. GRUR 1977, 96 – Dispositionsprogramm; GRUR 1978, 102 (103) – Prüfverfahren.

¹⁰⁰ *Mellius*; GRUR 1998, 843 (851); *Horns*, GRUR 2001, 12 (13), der für ein zeitgemäßes Verständnis des patentrechtlichen Technikbegriffes plädiert, ohne dieses aber eingehend zu konkretisieren; *Ohly*, CR 2001, 809 (812f.); *Anders*, GRUR 2001, 555 (556).

¹⁰¹ *Weyand/Haase*, GRUR 2004, 198 (200).

den technischen Charakter des Gegenstandes an sich durchschlagen, für den Patentschutz begehrt wird¹⁰².

Festgestellt werden muss, dass die juristische Auseinandersetzung in dieser Frage in hohem Maße ergebnisorientiert geführt wird¹⁰³. Dies gilt insbesondere für den in § 1 PatG enthaltenen Ausschlusskatalog¹⁰⁴. In der Diskussion um die Patentfähigkeit computerimplementierter Erfindungen werden in jüngerer Zeit Korrekturen der bisherigen Praxis eingefordert¹⁰⁵. So wird etwa von *Kiesewetter-Köbinger* daraufhin gewiesen, dass ein Programm alleine eine technische Aufgabe nicht zu lösen vermag und dies im Umkehrschluss nicht dazu führe, dass man dem das Programm ausführenden Computersystem ein erfinderisches Merkmal verleihen könne. Bei konsequenter Anwendung des Ausnahmekataloges des § 1 PatG muss dies seiner Auffassung nach folgerichtig zum Ausschluss der Patentfähigkeit des Gegenstandes „Computersystem“ führen, da es insoweit an der erfinderischen Tätigkeit mangelt¹⁰⁶.

Nach der Rechtsprechung der BGH ist im Rahmen der Prüfung der Patentfähigkeit einer Anmeldung die Feststellung des technischen Charakters der Erfindung notwendig, demzufolge ist die programmtechnische Verbindung mit einer Datenverarbeitungsanlage (Vorrichtung) erforderlich¹⁰⁷.

8.2.3 Genehmigungspraxis des DPMA

Im Rahmen der Überprüfung der Patentfähigkeit der auf Interoperabilität angewiesenen Softwaresysteme ist ferner die Genehmigungspraxis des DPMA einzubeziehen. Grundlage der Prüftätigkeit des DPMA sind die „**Richtlinien für die Prüfung von Patentanmeldungen**“¹⁰⁸. Sie bilden für den Anmelder erkennbar ab, welche Grundsätze die Prüfer in dem jeweiligen Verfahren anwenden werden. Zwar handelt es sich in dem konkreten Verfahren jeweils um eine Einzelfallprüfung, gleichwohl kommt den Prüfungsrichtlinien leitende Funktion zu und ermöglicht

¹⁰² *Anders*, GRUR 2004, 461 (462).

¹⁰³ Vgl. gleichlautende Regelung in Art. 52 Abs. 2 EPÜ

¹⁰⁴ Hierauf völlig zu Recht hinweisend, *Kiesewetter-Köbinger*, GRUR 2001, 185 (188).

¹⁰⁵ *Kiesewetter-Köbinger*, GRUR 2001, 185 (186 ff.).

¹⁰⁶ Vgl. auch *Anders*, GRUR 2004, 461 (463ff.) der anhand der Entscheidung „Tauchcomputer“ (*BGH* GRUR 1992, 430ff.) nachweist, dass bei genauerer Betrachtung die Analyse im Einzelfall zu dem Nachweis führen kann, dass nicht in jedem Fall ein vermeintlich allgemeines Prinzip in einer höchstrichterlichen Entscheidung enthalten ist. Zugleich hat in der Zwischenzeit eine intensive Diskussion um die Bedeutung nichttechnischer Merkmale, vor allem vor dem Hintergrund der Patenterteilungspraxis in den U.S.A eingesetzt, vgl. *Anders*, a.a.O.

¹⁰⁷ *BGH* GRUR 2002, 143 (145) – Suche fehlerhafter Zeichenketten. Gleichwohl wird es generell für möglich gehalten, dass eine Patentanmeldung, die vordergründig auf die Erlangung von Patentschutz für eine computerimplementierte Erfindung, sozusagen als trojanisches Pferd benutzt wird, um auf eine an sich nicht patentfähige Geschäftsmethode ein Patent zu erhalten, indem es als „vorrichtungsmäßig gekennzeichnetes Merkmal“ in eine industriell herstellbare und gewerblich einsetzbare Vorrichtung integriert wird, vgl. *Anders*, a.a.O. m.w.Nw. . Bemerkenswert ist, dass sich selbst in den U.S.A. eine kritische Auseinandersetzung mit allzu großzügiger Patentierungspraxis im Bezug auf die Schutzrechtserteilung von sog. „business methods“ eingestellt hat. Es würde aber nicht verwundern, wenn zumindest der Versuch unternommen würde, ähnliche Entwicklungen auch in Deutschland und der EU zu forcieren, was erste Patentanmeldungen erwarten lassen; so etwa das vom EPA unter der Patent Nr. EP 0 927 945 zu Gunsten der Fa. Amazon.Com.Inc. gewährte sog. „Amazon-One-Click-Patent“

¹⁰⁸ Siehe <http://www.dpma.de/formulare/p2796.pdf> Stand 1. März 2004 (letzter Abruf 16. September 2004).

ein Prognose über die zu erwartende Entscheidung; sie geben zugleich die Grundlinien der Genehmigungspraxis des DPMA wieder.

Unter Punkt 4.3 der Prüfungsrichtlinien des DPMA werden Anmeldungen, die DV-Programme oder Regeln enthalten, angesprochen. Da derartige Anmeldungen je nach Erfindungsgegenstand auf der Basis des IPC - Klassifikationssystems einer Prüfgruppe zugeordnet werden, darf die Wirkung der Prüfungsrichtlinien im Einzelfall nicht unterschätzt werden. Zwar besteht die IPC-Klasse G06 (Datenverarbeitung; Rechnen; Zählen), dies bedeutet jedoch nicht, dass Anmeldungen, die u.a. DV-Programme oder Regeln enthalten, zwangsläufig dieser Gruppe zugeordnet werden. Da kaum ein Technikbereich heutzutage ohne computerisierte Anwendungen auskommt, finden sich derartige Anmeldungen in nahezu jeder IPC-Klasse. Ein Prüfer, der nicht über spezielle oder vertiefte Informatikkenntnisse verfügt, wird deshalb die Entscheidungsfindung eng an den Prüfungsrichtlinien ausrichten¹⁰⁹. Die im Rahmen des Gutachtens im Rahmen der Bewertung der Patentfähigkeit der auf die Herstellung von Interoperabilität angewiesenen Softwaremodule einschlägigen Passagen werden im Folgenden auszugsweise wiedergegeben:

(4.3.1) Patenschutz für Erfindungen mit DV-Programmen, programmbezogenen Verfahren, Regeln oder dergleichen

Erfindungen, die ein DV-Programm, eine Rechen- oder Organisationsregel, sonstige Software-Merkmale oder ein programmbezogenes Verfahren enthalten, sind dem Patenschutz grundsätzlich zugänglich, sofern sie eine technische Lehre enthalten. Technisch ist eine Lehre zum planmäßigen Handeln unter Einsatz beherrschbarer Naturkräfte zur Erreichung eines kausal übersehbaren Erfolgs (*BGH GRUR 2000, 1007 – Sprachanalyseeinrichtung*).

(4.3.2) Programme, Regeln als solche

... Soweit sie (die in § 1 Abs. 2 lit. a) und b) genannten Gegenstände) dagegen zur Lösung eines konkreten technischen Problems Verwendung finden, sind sie – in diesem Zusammenhang – grundsätzlich patentfähig (*BGH CR 2002, 88f. – Suche fehlerhafter Zeichenketten*). Der Ausschluss gilt somit nicht für programmbezogene Erfindungen, d.h. in Programmen enthaltene oder als Verfahren oder Vorrichtungen formulierte Anweisungen

¹⁰⁹ Die Prüfungsrichtlinien des DPMA geben die Rechtsprechung des BGH in den angesprochenen Bereichen wieder und ergänzen diese teilweise um Beispiele. Als problematisch erweist sich hinsichtlich der Prüfung von Erfindungen nach 4.3.1 der Prüfungsrichtlinien in aller erster Linie die Dynamik des Softwaremarktes, auch wenn hinsichtlich der Werkzeuge für die Erstellung von Software eine zunehmende Standardisierung beobachtet werden kann. Dieser Umstand wirkt sich unmittelbar auf die notwendige Feststellung der Neuheit und der erfinderischen Tätigkeit aus. Im Rahmen der Prüfung des angemeldeten Patents hat ferner die Feststellung des „Stand der Technik“ zu erfolgen. Mit der wachsenden Zahl von Patenten für „computerimplementierte Erfindungen“ steigen die Anforderungen an Anmelder um Prüfer. Zwar versetzt ein umfassendes und gut beherrschbares elektronisches Recherchesystem den Patentprüfer in die Lage, den Stand der Technik festzustellen, jedoch ist dieser nur auf Entwicklungen begrenzt, die im Patentrechtssystem festgehalten sind. Da nach derzeitigem Kenntnisstand der Patenschutz im Bereich der Softwareindustrie nicht die vorherrschende Schutzform ist, stellt sich in diesem Fall die Frage, inwieweit im Rahmen des patentrechtlich relevanten Standes der Technik urheberrechtlich geschützte Werke zu berücksichtigen sind. Nach § 3 Abs. 1 Satz 2 PatG wird die Reichweite des „Stand der Technik“ umrissen. Der dort verwendete Begriff der „Kenntnisse“ ist umfassend angelegt und beschränkt sich nicht auf technische Lehren (*Keuckenschrijver* in: Busse Patentgesetz § 3 Rdnr. 74 m.w.Nw). Er erfaßt damit alle Kenntnisse, soweit sie vor dem für den Zeitrang der Anmeldung maßgeblichen Tag der Öffentlichkeit durch schriftliche oder mündliche Beschreibung, durch Benutzung oder in sonstiger Weise der Öffentlichkeit zugänglich gemacht worden sind (*Keuckenschrijver*, a.a.O. Rdnr. 76f.). Nicht maßgeblich ist in welcher Form, an welchem Ort oder vor wie langer Zeit die Öffentlichkeit Zugang erlangt hat (z.B. mündliche Mitteilungen in Vorträgen, in Hörfunk- und Fernsehsendungen, Gesprächen, Ausstellungen, Inverkehrbringen von Erzeugnissen, Anwendung von Verfahren sowie Informationen, die dem Internet entnommen werden können). Entscheidend ist, dass ein **Fachmann** des in Frage stehenden Gebiets die Möglichkeit gehabt haben muss, Kenntnis von diesen Informationen zu erlangen (*Keuckenschrijver*, a.a.O. Rdnr. 67, allgemein zur Feststellung des Standes der Technik, *Kraßer*, Patentrecht, S. 262ff.).

zum technischen Handeln. ... Dies gilt insbesondere auch für Programme zur Abarbeitung bestimmter Verfahrensschritte auf den herkömmlichen Gebieten der Technik (*BGH a.a.O.* – Suche fehlerhafter Zeichenketten).

(4.3.3.) Der technische Charakter der programmbezogenen Erfindung

Eine programmbezogene Erfindung hat technischen Charakter, wenn zur Lösung des der Erfindung zugrunde liegenden Problems von Naturkräften, technischen Maßnahmen oder technischen Mitteln Gebrauch gemacht wird oder wenn der Lösung technische Überlegungen zugrunde liegen (*BGH GRUR 2000, 498* – Logikverifikation).

Ob das der Fall ist, ist ... anhand der Merkmale des Patentanspruchs unter Berücksichtigung der ... Anmeldeunterlagen festzustellen.

Hierbei ist vom beanspruchten Gegenstand in seiner Gesamtheit auszugehen. ... Der Bezug zur Technik muss aus dem Patentanspruch hervorgehen (*BGH a.a.O.* – Logikverifikation).

Auf der Basis einer wertenden Betrachtung des im Patentanspruchs definierten Gegenstands ist festzustellen, ob ... ein ... Anmeldegegenstand den im § 1 Abs. 1 PatG vorausgesetzten technischen Charakter aufweist. ... Die Wertung der Technizität darf in ihrem Ergebnis aber nicht davon abhängen, ob der zu beurteilende Vorschlag neu und erfinderisch ist. Sie darf auch nicht einseitig darauf abstellen, was bekannt war und was nach der angemeldeten Lehre neuartig ist. Entscheidend ist, wie das, was nach der beanspruchten Lehre im Vordergrund steht, aus der Sicht des Fachmanns zum Anmeldezeitpunkt zu verstehen und einzuordnen ist (*BGH a.a.O.* – Logikverifikation). Eine Untersuchung ... zum Stand der Technik ... findet ... erst bei der Prüfung seiner Neuheit und der erfinderischen Tätigkeit statt.

(4.3.4) Verfahren/ Programm/ Schaltung/ Datenverarbeitungsanlage

Bei programmbezogenen Erfindungen ist der technische Charakter nicht davon abhängig, dass eine feste Schaltungsanordnung ... vorliegt. Derselbe Erfindungsgedanke, der einer solchen technischen Anordnung zugrunde liegt, kann auch als Verfahren und zwar im Zusammenwirken von Software mit programmierbarer Hardware patentfähig sein. Entscheidend ist, dass die Erfindung die Lösung des Problems unter Einsatz technischer Mittel oder technischer Überlegungen erfordert und lehrt.

Programmbezogene Erfindungen können daher auch dann technischen Charakter haben, wenn die zur Lösung herangezogenen technischen Mittel, ..., bereits bekannt sind. Unschädlich ist, dass die Elemente für sich genommen jeweils in bekannter Weise arbeiten. ...

Insbesondere weist ein Programm eine technische Lehre auf, wenn es in technische Abläufe eingebunden ist ...

(3.3.2.3) Neuheit (§ 3 PatG)

Bei der Prüfung auf Neuheit ist der beanspruchte Anmeldegegenstand mit dem Stand der Technik einzeln zu vergleichen (*BGH Zinkenkreisel*). Der Anmeldegegenstand ist neu, wenn er bei jedem einzelnen Vergleich mindestens ein Merkmal aufweist, das dem Stand der Technik fehlt. Maßgebend ist der Gesamthalt der jeweiligen Vorveröffentlichung also z.B. der Schrift, des Vortrags oder einer Vorbenutzung. ... Von Bedeutung ist ..., was der einschlägige Durchschnittsfachmann erkennen konnte.

(3.3.2.4.) Erfinderische Tätigkeit (§ 4 PatG)

... Dazu darf sie sich für den Fachmann nicht in nahe liegender Weise aus dem Stand der Technik ergeben. ...

Der Prüfer muss zunächst den Wissensstand ermitteln, der dem einschlägigen Durchschnittsfachmann vor dem für den Zeitrang der Anmeldung maßgeblichen Tag zur Verfügung stand. Dieser damalige Stand der Technik ist – in Verbindung mit dem Fachkönnen des Durchschnittsfachmannes – der Beurteilung zugrunde zu legen, ob sich der Erfindungsgegenstand für diesen Durchschnittsfachmann in nahe liegender Weise ergibt. Liegen die ermittelten Kenntnisse auf einem anderen Fachgebiet, ist es fraglich, ob diese dem Wissen des einschlägigen Durchschnittsfachmanns zugerechnet werden dürfen. ...

Ob erfinderische Tätigkeit vorliegt, hängt immer vom Einzelfall ab. ...

Indizien für das Vorliegen einer erfinderischen Tätigkeit sind z.B. eine sprunghafte Weiterentwicklung, die Überwindung technischer Vorurteile, vergebliche Bemühungen von Fachleuten, ...

Da gerade im Softwarebereich dem Internet als Informationsquelle besondere Bedeutung zukommt, muss beachtet werden, dass dieses nach Auffassung des BPatG als **Informationsquelle** zur Feststellung des Standes der Technik aus scheidet. Aktuell im Internet aufgefundene Informationen ließen für einen bestimmten Zeitpunkt der Vergangenheit ohne weiteren Nachweis nicht die Feststellung zu, dass sie damals schon im Internet eingestellt waren und dass ihre damalige Form mit der aktuellen inhaltlich identisch ist. Derartige Informationen werden in

diesem Zusammenhang somit erst dann relevant, wenn gleichzeitig belegt werden kann, ob und ggf. welche Veränderungen wurden¹¹⁰.

8.2.4 Patentschutz für „Computerimplementierte Erfindungen“ nach dem EPÜ

Die Basis für die Erteilung von Patenten durch das EPA ist das EPÜ¹¹¹. Es handelt sich nicht durch die Europäische Union gesetztes Recht, sondern vielmehr um einen multilateralen völkerrechtlichen Vertrag, der die „Europäisierung des Patentrechts“ als erklärtes Ziel verfolgt¹¹². Inhalt des Vertrages ist die Gründung der **Europäischen Patentorganisation** als zwischenstaatlicher Organisation, die kein Organ der Europäischen Union ist. Sie betreibt das EPA im München, das mit Wirkung für sämtliche Mitgliedsstaaten Patente erteilt. Diese Patente unterstehen wiederum der Hoheitsgewalt der einzelnen Mitgliedsstaaten¹¹³. In seiner derzeit geltenden Fassung lautet Art. 52 EPÜ in seinen hier interessierenden Passagen wie folgt:

Art. 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 Absatz 1 werden insbesondere nicht angesehen:
 - a) Entdeckungen sowie wissenschaftliche Theorien und mathematische Methoden; ...
 - c) Pläne, Regeln und Verfahren für gedankliche Tätigkeiten, für Spiele oder für geschäftliche Tätigkeiten sowie Programme für Datenverarbeitungsanlagen; ...

Absatz 2 steht der Patentfähigkeit der in dieser Vorschrift genannten Gegenstände nur insoweit entgegen, als sich die europäische Patentanmeldung oder das europäische Patent auf die genannten Gegenstände oder Tätigkeiten „als solche“ bezieht. Somit sind die Regelungen in § 1 PatG und Art. 52 EPÜ nahezu wortgleich formuliert. Danach besteht auch auf der Grundlage des EPÜ grundsätzlich das Verbot der Patentierung von Computerprogrammen. Mit Blick auf die Patentierung computerimplementierter Erfindungen ist die Situation betreffend der Auslegung der Vorschrift die gleiche, da im Jahr 1973 die Diskussion um die patentrechtliche Behandlung von Computerprogrammen eingesetzt hatte¹¹⁴.

8.2.4.1 Beispiele der Rechtsanwendungspraxis durch das EPA

Die Darstellung der Rechtsanwendungspraxis durch das EPA wird in diesem Gutachten auf die Analyse der Prüfungsrichtlinien des EPA sowie einer Reihe richtungweisender Entscheidungen der Beschwerdekammer des EPA begrenzt.

Für das Begriffsverständnis des „Computerprogramms als solchem“, ergeben sich aus dem EPÜ und den hierzu vorliegenden Materialien keine unmittelbaren Hinweise. Somit fehlt es dem

¹¹⁰ BPatG, Beschlüsse v. 9. Januar 2003, 17 W (pat) 47/00 sowie v. 17. Oktober 2002, 17 W (pat) 01/02 = BPatGE 46, 76 – Computernetzwerk-Information.

¹¹¹ Europäisches Patentübereinkommen vom 5. Oktober 1973 i.d.F. vom 10. Dezember 1998, Abl 1999, 1; umfangreiche Änderungen des EPÜ wurden auf der Münchener Diplomatischen Konferenz vom 20. bis 29. November 2000 beschlossen. Diese sind allerdings noch nicht in Kraft getreten. Hiervon betroffen ist auch der für die Patentfähigkeit von Computerprogrammen einschlägige Art. 52 EPÜ, der umformuliert, inhaltlich aber unverändert geblieben ist (vgl. *Nack/Phélip*, GRUR Int. 2001, 322 (323)).

¹¹² *Kraßer*, Patentrecht, S. 88f.

¹¹³ Siehe Art. 74 EPÜ.

¹¹⁴ Mikro- und makroökonomischen Implikationen der Patentierbarkeit von Softwareinnovationen – Endbericht, S. 167f.

für eine autonome Auslegung notwendigen Auslegungsmaterial. Allgemein wird jedoch von einer restriktiven Interpretation der Regelungen zum Patentierungsverbot ausgegangen¹¹⁵. Keine ausdrücklichen Regelungen finden sich in Bezug auf die Frage, wie die Herstellung von Interoperabilität gewährleistet werden kann.

Die Genehmigungspraxis des EPA spiegelt sich in den nach Art. 10 Abs. 2 lit. a EPÜ vom Präsidenten des EPA erlassenen internen Verwaltungsvorschriften wider, die bei der Prüfung einer Anmeldung eines europäischen Patents zugrunde gelegt werden. Zuletzt wurden die Grundsätze zur Behandlung von Geschäftsmethoden und computerbezogenen Erfindungen an die Rechtsprechung der Beschwerdekammern des EPA in der Aktualisierung der Prüfungsrichtlinien vom 31. August 2001 angepasst¹¹⁶ und ist durch die im Dezember 2003 vorgelegte Aktualisierung¹¹⁷ inhaltlich nicht verändert worden. Danach ist bei Prüfungsverfahren vor dem EPA von folgender Praxis auszugehen:

Richtlinien für die Prüfung im Europäischen Patentamt Kapitel IV, 2.3.6 (Stand Dezember 2003; Auszug)

... Zwar sind auch "Computerprogramme" in Art. 52(2) aufgeführt; hat der beanspruchte Gegenstand jedoch technischen Charakter, so ist er durch Art. 52(2) und (3) nicht von der Patentierbarkeit ausgeschlossen. ... Ein von einem Computerprogramm gesteuerter Datenverarbeitungsprozess ... umfasst immer physikalische Wirkungen, z.B. elektrische Ströme. Nach T 1173/97 sind solche normalen physikalischen Wirkungen alleine noch nicht ausreichend, um einem Computerprogramm technischen Charakter zu verleihen. Kann ein Computerprogramm beim Betrieb auf einem Computer aber eine weitere technische Wirkung hervorbringen, die über diese normalen physikalischen Wirkungen hinausgeht, so ist es nicht von der Patentierbarkeit ausgeschlossen, und zwar unabhängig davon, ob es allein oder als Aufzeichnung auf einem Datenträger beansprucht wird. Diese weitere technische Wirkung kann im Stand der Technik bekannt sein.

Eine weitere technische Wirkung, die einem Computerprogramm technischen Charakter verleiht, könnte z. B. in der Steuerung eines gewerblichen Verfahrens, in der Verarbeitung von Daten, die Gegenstände verkörpern, oder in der internen Funktionsweise des Computers selbst oder seiner Schnittstellen unter dem Einfluss des Programms zu finden sein und beispielsweise die Effizienz oder Sicherheit eines Verfahrens, die Verwaltung der erforderlichen Computerressourcen oder die Datenübertragungsgeschwindigkeit einer Kommunikationsverbindung beeinflussen. Somit kann ein allein oder als Aufzeichnung auf einem Datenträger oder in Form eines Signals beanspruchtes Computerprogramm als Erfindung im Sinne von Art. 52(1) betrachtet werden, wenn das Programm in der Lage ist, beim Betrieb auf einem Computer eine weitere technische Wirkung hervorzubringen, die über die normalen physikalischen Wechselwirkungen zwischen dem Programm und dem Computer hinausgeht.

8.2.4.2 Entscheidungspraxis der Beschwerdekammern des EPA

Die Beschwerdekammern des Europäischen Patentamts haben in einer Reihe von Entscheidungen aus den Bereichen der Datenverarbeitung und Geschäftsmethoden im Zusammenhang mit der Feststellung der Patentfähigkeit auf den Begriff der Technik Bezug genommen, es aber vermieden im Einzelnen die Voraussetzungen zu formulieren, die zur Patentfähigkeit führen. Dabei wird zwar auch die Bedeutung dieses Begriffs für die Abgrenzung von patentfähigen von nicht patentfähigen Erfindungen betont. Wie diese Abgrenzung letztlich zu erfolgen hat, ist in diesen Entscheidungen offen geblieben.

¹¹⁵ Ebenda, S. 168 m.w.Nw.

¹¹⁶ EPA ABI. 2001, 464.

¹¹⁷ Bislang ist lediglich die englische Fassung der Richtlinien für die Prüfung im Europäischen Patentamt elektronisch verfügbar (abrufbar unter http://www.european-patent-office.org/legal/gui_lines/pdf_2003/index_d.html, letzter Abruf 21. August 2004).

In der Entscheidung *Computerprogrammprodukt/IBM*¹¹⁸ ging die zuständige Beschwerdekammer davon aus, dass ein Programmprodukt, das potentiell einen vorgegebenen weiteren technischen Effekt bewirken könne, grundsätzlich nicht vom Patentschutz ausgeschlossen sei. Voraussetzung ist nach dieser Entscheidung, dass der technische Effekt über die normale physikalische Wechselwirkung zwischen Programm (Software) und der Datenverarbeitungsanlage (Hardware) hinausgeht, auf der es läuft. Bemerkenswert ist dabei, dass die Beschwerdekammer zwar betont, dass materiellrechtlich die Entscheidung auf der Basis des EPÜ zu treffen ist, bezieht aber zugleich in die Entscheidung ausdrücklich neuere Entwicklungen auf dem Gebiet des Patentschutzes für Computerprogramme in den USA und Japan sowie des TRIPS-Übereinkommens ein. Wird auf Rechtsentwicklungen aus anderen Rechtskreisen Bezug genommen, setzt dies an sich voraus, dass aus rechtsvergleichender Sicht die Übertragbarkeit der in einer fremden Rechtsordnung gewonnenen Ergebnisse zu überprüfen wäre, dass wegen der unterschiedlichen Systematik der Rechtsfindung in den angesprochenen Rechtsordnungen bzw. des internationalen Übereinkommens, nur unter großer Zurückhaltung erfolgen kann¹¹⁹. Entsprechende Erwägungen finden sich in der Entscheidung *Computerprogrammprodukt/IBM* nicht.

Beispielhaft für die Entscheidungspraxis sind folgende Entscheidungen der Beschwerdekammern des EPA¹²⁰:

Entscheidung der Beschwerdekammer des EPA	Technischer Charakter und Patentfähigkeit (+)
EPA, ABI. 1987, 14 - VICOM	Verfahren zur digitalen Bildbearbeitung , das durch besondere Konzeption den Rechenaufwand im Vergleich zu bekannten Verfahren beträchtlich vermindert
EPA, ABI. 1988, 19 - Koch & Sterzel	Röntgeneinrichtung , bei der mittels eines Programms die Röhren so gesteuert werden, dass sich gemäß einer Parameterprioritätsbeziehung eine optimale Belichtung bei hinreichender Sicherheit vor Überlastung der Röhren ergibt
EPA, ABI. 1990, 30 - Computerbezogene Erfindung/IBM	Verfahren, das mit Hilfe eines Computer und bestimmter in einem Speicher enthaltener Tabellen automatische Anzeigen von Zuständen bewirkt, in einer Vorrichtung oder einem System auftreten, Patentfähigkeit
EPA, ABI. 1990, 5 - Datenprozessornetz/IBM	Lehre zur Koordinierung und Steuerung der internen Kommunikation zwischen Programmen und Dateien, die in einem Datenverarbeitungssystem mit einer Vielzahl miteinander verbundener Prozessoren bei verschiedenen Prozessoren geführt werden
EPA, ABI. 1994, 557 – Editierbare Dokumentenform/IBM	Verfahren zur Umformung in einem Text enthaltener, für ein bestimmtes Textverarbeitungssystem geeigneter Steuerbefehle
EPA, GRUR Int. 1995, 909ff. - Universelles Verwaltungsprogramm / SOHEI	Computersystem zur Wahrnehmung mehrerer Arten von einander unabhängiger Verwaltungsaufgaben .
EPA, ABI. 1999, 609 - Datenstrukturpro-	Schutz eines Datenträgers , der ein Programm speichert, sofern das

¹¹⁸ EPA GRUR Int. 1999, 1053 – Computerprogrammprodukt/IBM (im Anmeldeverfahren wurde die Patentfähigkeit 2 von 3 angemeldeten Patentansprüchen noch verneint, da es sich um Computerprogramme i.S.d. Art. 52 Abs. 2 und 3 EPÜ handele); EPA T 208/84 VICOM, ABI. EPA 1987, 14 - computerbezogene Erfindung.

¹¹⁹ Rechtsprechung der Beschwerdekammern des EPA, 4. Aufl. 2001, S. 5f. (abrufbar über http://www.european-patent-office.org/legal/case_law/d/index.htm, letzter Aufruf 21. August 2004).

¹²⁰ Gesamtübersicht s. EPA, Rechtsprechung der Beschwerdekammern des Europäischen Patentamts, S. 2ff.

dukte	Computerprogramm auf einem Computer läuft oder in ihn geladen wird und eine technische Wirkung hervorbringt oder dazu in der Lage ist, der über die normale physikalische Interaktion zwischen Programm und Computer hinausgeht.
Entscheidung der Beschwerdekammer des EPA	Technischer Charakter und Patentfähigkeit (-)
EPA, ABI. 1990, 12 Zusammenfassen und Wiederauffinden von Dokumenten/IBM	Verfahren zur Zusammenfassung eines Dokuments, Speichern der Zusammenfassung und deren Wiederauffinden. Die Bereitstellung eines Regelwerks für ein in erster Linie nach administrativen Gesichtspunkten ablaufendes Informationswiederauffindungsverfahren, welches mit Hilfe herkömmlicher Computerhardware realisiert werden könne, verleiht diesem keinen technischen Charakter. Es wird nicht die physikalische Erscheinung, sondern nur der Informationsgehalt von Dokumenten wiedergegeben.
EPA, ABI. 1990, 384 - Textverarbeitung/IBM	Verfahren zur automatischen Erkennung und Substitution sprachlicher Ausdrücke , die gemäß einer entsprechenden Liste oberhalb eines vorgegebenen Verständlichkeitsniveaus liegen. Da die gespeicherten Informationen rein abstrakter, sprachlicher Natur seien, ist die Gesamtwirkung des Verfahrens nicht technisch
EPA, ABI. 1991, 566 - Schriftzeichenform/Siemens	Verfahren zur Darstellung von Schriftzeichen in einer fest vorgegebenen von mehreren möglichen Zeichenformen auf einem Datensichtgerät. Dem Wesen nach handele es sich nicht um ein technisches Betriebsverfahren für eine Datenverarbeitungsanlage, sondern eine Programmidee, die keine über die bloße Informationsverarbeitung hinausgehende Wirkung auslöse.
EPA, RBK 1993, 17 – Erzeugung von Computerprogrammbausteinen	System zur Schaffung konkreter Computerprogramme aus Bausteinen oder Modulen. Die Kammer entschied, dass zwar die Effizienz des Programmiervorgangs erhöht werden, hierdurch aber der Computer aus technischer Sicht auf eine im wesentlichen neue Weise arbeite.
EPA, ABI. 2001, 441 - Pensionssystem	Verfahren zur Kontrolle eines Pensionskassensystems durch Verwaltung mindestens eines angeschlossenen Arbeitgeberkontos. Die Kammer entschied, dass es sich um eine nicht patentierbare Geschäftsmethode handele. Die Beschwerdekammer nahm auf die Entscheidung Datenstrukturprodukt (ABI. 1999, 609) Bezug und stellt fest, dass das Verfahren, auch wenn es technisch sei oder in anderen Worten technischen Charakter aufweise, trotzdem ein Verfahren für eine geschäftliche Tätigkeit als solche sei. So wie die Ansprüche formuliert wurden ging die Erfindung nicht über ein Verfahren für geschäftliche Tätigkeiten als solche hinaus; es fehlte an der notwendigen erfinderischen Tätigkeit in Bezug auf das beanspruchte Verfahren ¹²¹ . Der Vorrichtung wurde dagegen technischer Charakter zuerkannt ¹²²
EPA, GRUR Int. 2003, 852 – Zwei Kennungen/COMVIC	Bei einer Erfindung, die aus einer Mischung technischer und nichttechnischer Merkmale besteht und als Ganzes technischen Charakter aufweist, sind in Bezug auf die Beurteilung des Erfordernisses der erfinderischen Tätigkeit alle Merkmale zu berücksichtigen, die zu diesem technischen Charakter beitragen , wohingegen Merkmale, die keinen solchen Beitrag leisten, das Vorliegen erfinderischer Tätigkeit nicht stützen können. Die zu lösende technische Aufgabe ist zwar nicht so zu formulieren, dass sie Lösungsansätze enthält oder die Lösung teilweise

¹²¹ Zur Schwierigkeit den Entscheidungen der Beschwerdekammern des EPA klare Leitlinien entnehmen zu können, *Kraßer*, Patentrecht, S. 152f.

¹²² Ausführliche Analyse der Entscheidung bei *Kraßer*, Patentrecht, S. 170.

	vorwegnimmt, doch scheidet ein Merkmal nur deshalb, weil es im Anspruch vorkommt, nicht automatisch für die Formulierung der Aufgabe aus. Insbesondere wenn der Anspruch auf eine Zielsetzung auf einem nichttechnischen Gebiet verweist, darf diese Zielsetzung bei der Formulierung der Aufgabe als Teil der Rahmenbedingungen für die zu lösende technische Aufgabe aufgegriffen werden, insbesondere als eine zwingend zu erfüllende Vorgabe.
--	---

Patentfähig ist eine computerimplementierte Erfindung danach dann, wenn sie in der Gesamtsicht technischen Charakter aufweist. Nicht ausreichend ist die Implementierung einer Geschäftsidee als Programm, da in diesen Fallgestaltungen lediglich der bestimmungsgemäße Gebrauch des Computers umschrieben wird¹²³.

Insgesamt ist die Entscheidungspraxis nach Auffassung von *Kraßer*¹²⁴ noch immer durch eine gewisse Großzügigkeit geprägt. Beklagt wird gleichzeitig, dass die Entscheidungspraxis durch erhebliche Unterschiede gekennzeichnet ist, welche Kriterien der Prüfung letztlich zugrunde gelegt werden müssen¹²⁵, bis hin zu der Entscheidung, ob der technische Charakter einer computerimplementierten Erfindung aus der Tatsache abgeleitet werden könne, dass ihr Auffinden technische Überlegungen erfordere (vgl. Universelles Verwaltungsprogramm/ SOHEI).

In den Prüfungsrichtlinien des EPA wird auf die Entscheidungen der Beschwerdekammer des EPA Bezug genommen. Diese Vorgehensweise wird in der Literatur durchaus kritisch bewertet und der Vorwurf formuliert, dass die Erfassung des Technikbegriffes auf ein gefühlsmäßiges Erfassen reduziert werde. Durch subjektive bedingte Abweichungen würden damit letztlich unterschiedliche Ergebnisse bei vergleichbaren Gegenständen möglich, ein Ergebnis, dass unter dem Blickwinkel der Rechtssicherheit als problematisch zu erachten ist¹²⁶. Wie *Kiesewetter-Köbinger* zutreffend feststellt, ist die Praxis der Patentierung von computerimplementierten Erfindungen mit vielfältigen Problemen behaftet, die es erwarten lassen, dass in dem Moment, in dem auf der Seite der Software-Entwickler die „Schmerzschwelle“ überschritten wird, eine Welle von Einspruchs- und Nichtigkeitsverfahren zu erwarten sein wird¹²⁷.

8.2.5 Zusammenfassung der Entscheidungspraxis bei „computerimplementierten Erfindungen

Eine Auswertung der Praxis, bezogen auf konkrete Softwaretechnologien ergibt danach folgendes Bild¹²⁸:

¹²³ *Anders*, GRUR 2001, 555 (558f.) unter Hinweis auf die Entscheidung „Pensionssystem“.

¹²⁴ *Kraßer*, Patentrecht, S. 148; auch auf der Grundlage der jüngeren Entscheidungspraxis des EPA sich kritisch äußernd, *Keuckenschrijver* in: Busse Patentgesetz § 1 Rndnr. 75.

¹²⁵ *Kraßer*, Patentrecht, S. 150.

¹²⁶ *Tauchert*, Zum Begriff der „technischen“ Erfindung, JurPC Web-Dok. 28/2002, Abs. 17, der diese Vorhensweise wie folgt charakterisiert: „etwa: es ist wie bei einem Kamel, es zu beschreiben fällt schwer, aber wenn man es sieht, erkennt man es sofort“.

¹²⁷ *Kiesewetter-Köbinger*, GRUR 2001, 185 (193).

¹²⁸ Mikro- und makroökonomischen Implikationen der Patentierbarkeit von Softwareinnovationen – Endbericht, S. 180f. Nr. 255 - 257.

Softwaretechnologie	Technisch im patentrechtlichen Sinne	Nicht technisch im patentrechtlichen Sinne	unklar
Steuerungs- und Regelungstechnik	x		
CAD/CAM	x		
Digitale Signalbearbeitung	x		
Betriebssystem	x		
Hilfeprogramm	x		
Datenkompression	x		
Kundenmanagement	x		
Textverarbeitung		x	
Tabellenkalkulation		x	
Datenverschlüsselung		x	
Programmierwerkzeuge		x	
Authentifizierung		x	
Zeitreihenanalyse		x	
Verwaltungssoftware			x

Zusammenfassend lässt sich feststellen, dass sowohl BGH, als auch EPA im Laufe zunehmender Befassung mit computerimplementierten Erfindungen die Anforderungen an die Eröffnung des Patenschutzes für computerimplementierte Erfindungen abgesenkt haben¹²⁹. Mit dieser Entwicklung ist zugleich eine Neubestimmung des patentrechtlichen Technikbegriffes einhergegangen. Nicht ohne weiteres kann man in den Fällen computerimplementierter Erfindungen von dem Einsatz beherrschbarer Naturkräfte ausgehen, den das Patentrecht voraussetzt. Daher wird z.B. gefordert, man müsse von einem erweiterten Technikverständnis¹³⁰ bzw. einer liberal wertenden Auslegung des Technizitätsbegriffs¹³¹ ausgehen bzw. dem Begriff neben Materie und Energie auch die Information zuzurechnen¹³². Dem wird u.a. kritisch entgegengehalten, dass dieser eingeschlagene Weg in der letzten Konsequenz zur Patentierung von mathematischen Methoden führen könne¹³³. Aus der jüngeren Rechtsprechung des BGH zur Patentfähigkeit computerimplementierter Erfindung lässt sich entnehmen, dass der nicht jedwede in com-

¹²⁹ Als besonders problematisch erscheint die Mischung von technischen und nichttechnischen Merkmalen im Rahmen der Prüfung der erfinderischen Tätigkeit; vgl. hierzu Anders, GRUR 2004, 461 (462) u.a. mit Hinweis auf EPA, ABl. 2001, 441 – Pensionssystem; EPA 2003, 352 – Zwei Kennungen/COMVIC. Im Rahmen der Ermittlung des technischen Beitrags nach §§ 1 Abs. 1, 4 PatG ist der Gegenstand in seiner Gesamtheit unter Einschluß der an sich nichttechnischen Merkmale zu betrachten; zuletzt BPatG, Beschluss v. 10. Mai 2004 20 W (pat) 314/02, S. 10f. (www.softwarepatentnews.de/pdf/20314_02.pdf, letzter Abruf 15. Nov. 2004).

¹³⁰ Pierson, S. 198ff., 204ff., 212f.

¹³¹ Goebel, FS Nirk, S. 378.

¹³² Kraßer, Patentrecht, S. 164 m.w.Nw.

¹³³ Winischhofer, Computersoftware und Patentrecht, S. 86.

putergerechte Anweisungen gekleidete Lehre als patentierbar erachtet werden kann, wenn sie nur irgendwie über die Bereitstellung der Mittel hinausgeht, die die Nutzung als Datenverarbeitungsprogramm erlauben¹³⁴. Die Frage, ob hieraus ein allgemeiner Trend zur Verschärfung der Anforderungen an die Patentierbarkeit von computerimplementierten Erfindungen ablesbar ist, muss mit Zurückhaltung beantwortet werden, zumal die Entscheidungen, die zur Verneinung der Patentfähigkeit geführt haben, häufig Fälle betrafen, in denen der Versuch unternommen wurde für Geschäftsmethoden Patentschutz zu erlangen¹³⁵.

Im Bereich der gewerblichen Nutzung von Erfindungen kennt das Patentrecht, im Gegensatz zum Urheberrecht in § 69e UrhG, kein Recht des Benutzers auf Dekompilierung¹³⁸. Durch die erhebliche Ausweitung der als technisch zu qualifizierenden Gegenstände, die im wesentlichen computerimplementierte Erfindungen betrifft, ist eine Entwicklung in Gang gesetzt worden, die die Herstellung der Kommunikationsfähigkeit zwischen Computerprogrammen erschwert, wenn nicht unmöglich macht. Welche Instrumente sinnvoller Weise zum Einsatz gelangen können, um innerhalb des Patentrechts die Frage nach den Voraussetzungen beantworten zu können, unter denen eine Dekompilierung ermöglicht werden könnte, setzt einen kurzen Blick auf die Patentfähigkeit der betroffenen Systeme voraus.

8.2.6 Patentfähigkeit von Softwaresystemen

Die vorgehend dargestellten Prinzipien sollen nachfolgend auf die in Kapitel 5 dargestellten Softwaresystem übertragen werden. Dabei wird der Begriff Software auf der Grundlage DIN 44300 (1988) definiert. Ein Computerprogramm ist danach als eine nach Regeln der verwendeten Sprache (Programmiersprache) festgelegte syntaktische Einheit aus Anweisungen und Vereinbarungen zu verstehen, die die zur Lösung einer Aufgabe notwendigen Elemente umfasst.

¹³⁴ So *Keuckenschrijver* in: Busse Patentgesetz § 1 Rdnr. 52 m.w.nN.

¹³⁵ So etwa im Fall EPA, ABl. 2001, 441 – Pensionssystem. Besondere Aufmerksamkeit hat das sog. Amazon 1-Click-Patent auf sich gezogen. Sowohl vom US Patent and Trademark Office (US Pat.No.5,960,441), als auch vom EPA (EP0902381) wurde der Fa. Amazon das sog. 1-Click-Patent erteilt. Der Patentanspruch ist wie folgt formuliert: „Method and system for placing a purchase order via a communications network“. Unter dem Dach einer computerimplementierten Erfindung wurde in diesem Fall zugleich eine Geschäftsmethode patentiert. Gegenstand des Patentes ist ein Verfahren, das dem Kunden einen Online-Einkauf mittels eines einzigen Klicks während der Online-Nutzung einer Verkaufsplattform gestattet. Nachdem der Kunde sich bei amazon.com angemeldet und sein persönlichen Daten, einschließlich der für den Zahlungsvorgang notwendigen Informationen, hinterlegt hat, werden diese mittels eines Cookie auf dem Rechner des Nutzers gespeichert. Amazon.com erlangt hierdurch die Möglichkeit im Onlinestatus sich mit diesem Profil in Verbindung zu setzen. Legt der Kunde Waren in seinem virtuellen Einkaufswagen ab, ist lediglich ein Klick eine bestimmte Schaltfläche notwendig, um den Bestellvorgang auszulösen. Vermittels des Cookie findet amazon.com die Nutzerdaten und kann ohne weitere Aktivitäten des Nutzers die Bestellung abwickeln.

¹³⁸ Zur Entstehungsgeschichte *Blocher* in: Europäisches Urheberrecht, Art. 6 Software-RL Rdnr. 1ff. der auf den schon damals bestehenden Konflikt zwischen großen Standardsoftwareherstellern und kleineren und mittleren Softwareherstellern um die Frage der Gefahr der Monopolisierung von Schnittstellen hinweist.

Weiterhin hat die WIPO Mustervorschriften entwickelt, die eine Definition des Begriffs Software enthalten¹³⁹.

8.2.6.1 Betriebssysteme

Als Betriebssystem wird ein Computerprogramm verstanden, das die Ausführung von Anwendungsprogrammen steuert. Es ist sozusagen die Schnittstelle zwischen Anwendung und Hardware (vgl. 5.1). Für sich genommen kann ein Betriebssystem wegen des Verbots der Patentierung von Software „als solcher“ keinen Patentschutz erlangen. Nach dem im technischen Teil des Gutachtens zugrunde gelegten Verständnis, kommt daher als Schutzrecht für Betriebssysteme allein das Urheberrecht in Frage, da es an dem für den Patentschutz erforderlichen technischen Charakter fehlt, wenn lediglich Anwendungen (Software) miteinander verbunden werden. Dient das Betriebssystem der effizienten Nutzung von Rechnerressourcen, kommt Patentschutz dann in Betracht, wenn es z.B. zu einer Optimierung der eingesetzten Hardware führt. Auch Betriebssysteme können damit grundsätzlich dem Patentschutz unterliegen¹⁴⁰.

Die Frage des Patentschutzes von Betriebssystemen kann daher nur einzelfallbezogen beantwortet werden.

8.2.6.2 Kommunikationssysteme

Nach 5.2.1 dienen Kommunikationssysteme dem rechnerübergreifenden Austausch von Nachrichten durch ein Rechnernetz. Erforderlich ist hierfür die Bereitstellung von **Kommunikationssoftware**. Im Kern geht es somit um die Verbindung separater Rechner durch Vermittlungsknoten.

In diesem Fall ist zweifelhaft, ob der Einsatz der Kommunikationssoftware Wirkungen erzeugt, die über die normale physikalische Interaktion zwischen Programm und Computer hinausgehen. Ist dies nicht der Fall, scheidet Patentschutz aus. Wird dagegen Kommunikationshardware gesteuert, wie z.B. Ethernet und ISDN, kann ein technischer Charakter durchaus bejaht werden.

Je nach Einzelfall besteht daher die Möglichkeit, dass auch Kommunikationssysteme dem Patentschutz unterliegen können. Da jedoch, wie unter 5.2.2 und 6.4 dargestellt, Kommunikationsprotokolle im Regelfall als Standards genormt wurden, ist im Regelfall davon auszugehen, dass kein Patentschutz besteht.

8.2.6.3 Verteilte Systeme

Verteilte Systeme sind ebenfalls Rechnernetze, die durch zusätzliche Softwareschichten (Middleware) organisiert werden. Das derartige Verfahren dem Patentschutz zugänglich sind, zeigen erste erteilte Patente. So wurde vom EPA der Fa. Microsoft Verfahrenspatent über das verteilte Bearbeiten von Dateien erteilt (Method and system for open file caching in a computer system)¹⁴¹. Es handelt sich um einen proprietären Kommunikationsstandard der im Rahmen einer Rechnernetz zur Anwendung gelangt.

¹³⁹ Text s. *Marly*, Softwareüberlassungsvertrag, Rdnr. 8; siehe auch GRUR Int. 1978, 286.

¹⁴⁰ Konkret erteilte europäische oder deutsche Patente wurden nicht ermittelt.

¹⁴¹ EP0438571, US Pat.No.5,261,051; <http://patent.tange.dk/wiki/EP/4/3/EP438571.html> (letzter Abruf: 23. September 2004).

8.2.6.4 Anwendungssysteme

Unter Anwendungssystemen werden Systeme verstanden, die Softwaremodule enthalten (sog. Applikationssoftware). Zu dieser Kategorie zählen u.a. Textverarbeitungsprogramme, aber auch Programme zur Steuerung von Maschinen (vgl. 5.4.1). Anwendungssysteme können daher lediglich als Bestandteil einer computerimplementierten Erfindung Patentschutz erlangen.

8.2.6.5 Trusted Computing, Nexus

Computersysteme, Computer, Peripheriegeräte, Telekommunikationsendgeräte u.a.m. sind seit geraumer Zeit latenten Gefährdungen durch Spam-Attacken, Viren, Trojaner und andere Schadprogramme ausgesetzt. Hierdurch entstehen den betroffenen Personen, Organisationen und Unternehmen erhebliche finanzielle Schäden¹⁴². Eine Reihe von in der Mehrzahl US-amerikanischer Unternehmen hat sich deshalb in der Trusted Computing Group zusammengeschlossen¹⁴³. Entwickelt werden Spezifikationen mit denen Hardwarebauteile ausgerüstet werden. Eine dieser Spezifikationen ist das Trusted Platform Module (TPM), mit dem z.B. Motherboards ausgerüstet werden¹⁴⁴. Die Planung und Standardisierung weiterer Anwendungsmöglichkeiten und insb. Schnittstellen ist noch nicht abgeschlossen¹⁴⁵.

Angesprochen werden u.a. interne Verbindungen und Geräteanbindungen, unter Anwendung softwarebasierte Verfahren und Methoden¹⁴⁶. In den U.S.A. sind für den Bereich des Trusted Computing Patente erteilt worden.

Bspl. US Pat.No. 6,782,349

“Method and system for updating a root of trust measurement function in a *personal computer*

Abstract

A method and system for *updating a root* of trust measurement (RTM) function in a *personal computer* is disclosed. The RTM function is located in a boot block of the *personal computer*. The method and system comprise initializing a request to update the RTM function and unlocking the boot block based on an authentication process. The method and system further includes updating the RTM function. Through the use of the method and system in accordance with the present invention, the RTM function in a *personal computer* is updated in a manner that ensures that the update is authentic.”

8.2.6.6 Kryptoalgorithmen, Sicherheitsprotokolle und Sicherheitsinfrastrukturen

Algorithmen sind Rechenregeln und unterliegen folglich dann nicht dem Patentschutz, wenn sie ohne Bezug zu einer physischen Umgebung definiert werden¹⁴⁷. Umfasst er technische Prozesse, kann er dem Patentschutz unterliegen¹⁴⁸.

¹⁴² Allgemein zum Thema Trusted Computing s. *Schallbruch*, DuD 2004, 519f.

¹⁴³ Siehe im Einzelnen S. 30f; die Auflistung der aktuellen Mitglieder der TCG findet sich unter <http://www.trustedcomputinggroup.org/about/members> (letzter Abruf 4. Oktober 2004).

¹⁴⁴ *Brandl/Rosteck*, DuD 2004, S. 529 (530).

¹⁴⁵ *Brandl/Rosteck*, a.a.O.

¹⁴⁶ Eine Auflistung von Sicherheitfunktionen geben *Brandl/Rosteck*, DuD 2004, S. 529 (531).

¹⁴⁷ Z.B. EP 0482154.

¹⁴⁸ Vorschlag für eine Richtlinie des Europäischen Parlaments und des Rates über die Patentierbarkeit computerimplementierter Erfindungen, KOM (2002) 92 endg., S. 18

8.2.6.6.1 Algorithmus

Ein Algorithmus ist die Beschreibung eines Datenverarbeitungsverfahrens, dem in der Informatik grundlegende Bedeutung zukommt¹⁴⁹. Ein Algorithmus stellt eine Beschreibungsmöglichkeit für die schrittweise Ausführung einer Lehre dar, ebenso wie ein Programm für eine Datenverarbeitungseinrichtung. Mit einem Algorithmus oder einem Programm für eine Datenverarbeitungseinrichtung können technische und nicht-technische Lehren beschrieben werden¹⁵⁰.

Der Richtlinienentwurf definiert folglich die Funktion des "Algorithmus" im weitesten Sinne als jede detaillierte Handlungsfolge, die der Erfüllung einer bestimmten Aufgabe dient¹⁵¹. Da ein Computer eine Aufgabe nur dann lösen kann, wenn ihm der Problemlösungsweg durch eine detaillierte Folge von Befehlen mitgeteilt wird, liegt die Bedeutung von Algorithmen für die Informatik auf der Hand: Ohne Algorithmus gibt es kein Programm! - Der Algorithmusbegriff ist allerdings kein Exklusivbegriff der Informatik, sondern erfasst auch Methoden zur nicht DV-gestützten, d.h. konventionellen Lösung von Aufgabenstellungen (z. B. Montageanleitungen, konventionelle Maschinen etc.)¹⁵².

Aus den BGH-Entscheidungen „Logikverifikation“ und „Sprachanalyseeinrichtung“ wird abgeleitet, dass auch algorithmische Softwareerfindungen grundsätzlich patentfähig sind¹⁵³. Softwareerfindungen müssen nämlich nicht stets unmittelbar in physikalische Kausalketten eingebunden sein, sondern es reicht aus, wenn der Patentgegenstand einen Lösungsvorschlag in einem Prozess betrifft, der auf den unmittelbaren Einsatz von beherrschbaren Naturkräften verzichtet.

8.2.6.6.2 Der Viterbi-Algorithmus

Ein weiteres Beispiel ist die Entscheidung "**Viterbi-Algorithmus**" des BPatG vom 25.03.1996¹⁵⁴. Wie viele andere Patente im Bereich computerprogrammbezogener Erfindungen wurde auch dieses Patent aufgrund einer U.S.-amerikanischen Priorität durch das DPMA erteilt¹⁵⁵. Beschrieben wird ein Algorithmus, mit dem verschiedene Funktionen in Nachrichtempfängern (wie Demodulieren, Decodieren, Entzerren) durchgeführt werden können. Das übertragene Signal wird auf Störungen analysiert. Bisher wurde die Entscheidung, ob die Übertragung fehlerfrei war, auf binäre Weise getroffen, sog. „harte“ Entscheidung. Der neue Algorithmus sieht eine „Soft-Decision“ vor; dies ist eine auf der binären Entscheidung beruhende nach dem Algorithmus geprüfte Entscheidung, ob die Übertragung fehlerfrei war. Das BPatG hat den weiterentwickelten Algorithmus als einen mit technischem Inhalt qualifiziert, da er zur Verarbeitung von technischen Signalen diene und damit eine Lehre zum technischen Handeln gebe. Dabei stehe es dem technischen Charakter nicht entgegen, dass der Beitrag zum Stand der Technik ausschließlich in der Bereitstellung mathematischer Regeln bestehe und damit auf nichttechnischem Gebiet liege. Bezugnehmend auf den Seitenpuffer-Beschluss hinge die Frage der Technizität nicht davon ab, ob die Lehre neu, fortschrittlich und erfinderisch sei. Durch die Aufnahme der Zweckangabe in den Patentanspruch „zum Empfang von über einen gestörten Kanal übertragenen Signalen“ stehe der Algorithmus in so enger Beziehung zu technischen

¹⁴⁹ Ausdrücklich Patenschutz fordernd *Horns*, GRUR 2001, 1 (7).

¹⁵⁰ *BPatG*, GRUR 1996, 866 (867) - Viterbi-Algorithmus (= CR 1996, 665 m. Anm. Betten) - und *BGH*, GRUR 1980, 849 (851) – Antilockiersystem.

¹⁵¹ Vorschlag für eine Richtlinie des Europäischen Parlaments und des Rates über die Patentierbarkeit computerimplementierter Erfindungen, KOM (2002) 92 endg., S. 8.

¹⁵² *Pierson*, JurPC Web-Dok. 18/2004 (Algorithmus).

¹⁵³ *Horns*, GRUR 2001, 1 (10).

¹⁵⁴ *BPatGE* 36, 174.

¹⁵⁵ US Pat.No.5,181,209, EP0391354B1, DE3910739C3.

Vorgängen, dass er inhaltlich auf technische Größen festgelegt sei und unmittelbar umschrieben werde, welche Signale mit dem Algorithmus schrittweise verarbeitet werden¹⁵⁶.

Wie die Praxis zeigt, ist ein Algorithmus nicht nur eine abstrakt-allgemein gehaltene Regel, sondern wird auch für konkrete Anweisungen, Computerprogramme oder technische Handlungsanweisungen verwendet¹⁵⁷. Am Ende wird die Entscheidung über die Gewährung von Patentschutz im Einzelfall davon abhängen, ob das beanspruchte Patent in der Gesamtsicht technischen Charakter aufweist.

8.2.6.6.3 Kryptographische Algorithmen

Sog. „Encryption Algorithmus“, die die Sicherheit der Datenübertragung gewährleisten und gespeichert Daten vor unberechtigtem Zugriff durch Dritte schützen sollen, sind auch in Europa bereits patentiert worden. Eine programmierbare Datenverschlüsselungsmaschine für einen Algorithmus des erweiterten Verschlüsselungsstandards wurde vom EPA am 15. September 2004 zugunsten einer U.S.-amerikanischen Anmelderin, der Fa. Analog Devices Inc. erteilt¹⁵⁸, auch hier aufgrund U.S.-amerikanischer Priorität. Ein weiteres Beispiel ist das patentierte blockorientierte Verschlüsselungsverfahren **IDEA** (International Data Encryption Algorithm), welches die meisten Pretty-Good-Privacy (PGP)-Varianten verwenden können. Patentinhaber sind die ETH Zürich und das schweizerische Unternehmen Ascom Systec AG. Entsprechende Patente wurden in Europa, U.S.A. und der Schweiz erteilt¹⁵⁹. Es handelt sich nach den Patentschriften um ein „Device for the conversion of a digital block and use of same“ und wird im „Abstract“ der U.S.-amerikanischen Patenterteilungsschrift zusammengefasst als: „The device (12) comprises nine encryption stages (61.1, 61.2, 69), the first eight of which are constructed identically. It further comprises a key subblock generation unit (63), an input unit (21), and an output unit (79). It serves for the block-by-block encryption of a plaintext (X) proceeding from a message source (11) into a ciphertext (Y) to be delivered on a transmission line (13), wherein a secret key block (Z) is inputted beforehand via a secure channel 17. The encryption is effected in a step-by-step and parallel manner for four subblocks (X.sub.1 -X.sub.4 ; W.sub.11 -W.sub.14 ; W.sub.21 -W.sub.24 ; W.sub.81 -W.sub.84 ; Y.sub.1 -Y.sub.4). Every encryption stage (61.1, 61.2, 69) comprises four first inputs (25-28; 35-38), six and four second inputs (29, 30, 32, 33, 49, 52; 129, 130, 132, 133), respectively, and four outputs (75-78). A total of fifty-two key subblocks (Z.sub.1 -Z.sub.52) which are formed from the key block (Z) are connected to the second inputs. The device (12) can also serve, without being altered, for the decryption of an incoming ciphertext (Y). Different key subblocks need only be connected to the second inputs for this purpose.“¹⁶⁰

Der IDEA beruht auf mathematischen Eigenschaften einer algebraischen Struktur. Es werden drei unterschiedliche, nicht "kompatible" Abbildungen aus drei verschiedenen solcher Gruppen miteinander verknüpft. IDEA findet in populären Anwendungen wie **PGP** und einigen Freeware-Programmen Verwendung¹⁶¹.

¹⁵⁶ <http://www.softwarepatentschutz.de/index.php?id=rechtsprechung&subcat=Viterbi-Algorithmus> (letzter Abruf 23. September 2004).

¹⁵⁷ *BPatG*, GRUR 1996, 866 (867) – Viterbi-Algorithmus = CR 1996, 665 m. Anm. *Betten*; *BPatGE*, CR 1997, 616.

¹⁵⁸ EP1456994.

¹⁵⁹ EP0482154B1; US Pat.No.5,214,703, PCT/CH91/00117.

¹⁶⁰ <http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=/netahtml/srchnum.htm&r=1&f=G&l=50&s1=5214703.WKU.&OS=PN/5214703&RS=PN/5214703> (letzter Abruf 23. September 2004).

¹⁶¹ <http://mitglied.lycos.de/cthoeing/algo/idea.htm> (Auszug; letzter Abruf 23. September 2004).

Für den Bereich des „**Trusted Computing**“ (TC) ergeben sich hieraus naturgemäß weit reichende Konsequenzen, soweit es sich um kommerzielle Nutzung handelt. Allein im Fall nicht-kommerzieller Nutzung wird auf von den Patentinhabern regelmäßig auf die Zahlung von Lizenzgebühren verzichtet. Unternehmen oder Entwickler, die Anwendungen für TC-Verfahren entwickeln sind deshalb gezwungen, eine Lizenzvereinbarung abzuschließen, wobei nach derzeitiger Rechtslage kein Recht auf Dekompilierung bzw. Offenlegung des Quellcodes besteht. Als Ausweg besteht nur die Möglichkeit Mitglied der „Trusted Computing Group“ zu werden und die Möglichkeit der Einräumung der Offenlegung von Informationen nach Art. 15 der Satzung der Trusted Computing Group zu erhalten¹⁶².

8.2.6.7 Konvertierungsprogramme/Nachrichten-Broker

Der Austausch von Informationen zwischen Softwaremodulen kann mittels der Verwendung von Konvertierungsprogrammen erfolgen, d.h. Programme, mit denen man Daten in andere – auch plattformfremde – Datenformate umwandeln kann. Sie sind häufig in andere Anwendungsprogramme integriert (Importfunktion, Exportfunktion). Die informationstechnischen Einzelheiten wurden unter 6.1 dargestellt.

Zur Verdeutlichung der Problemstellung soll folgendes Beispiel dienen:

Bspl: Konvertierungsprogramm¹⁶³

Aufgabe des Konvertierungsprogramms ist die Zerlegung eines im RTF-Format gespeicherten Berichts entsprechend dessen Gliederungsstruktur, die Umwandlung in das HTML-Format und die Vernetzung der erzeugten Berichtsseiten. Es erfolgt eine Portierung von Unix nach Windows NT. Strukturelle Verweise werden am Anfang einer Berichtseite in einer Steuerleiste zusammengefasst. Inhaltliche Querverweise können an jeder Stelle innerhalb des Berichts erscheinen. Die nachfolgende Abbildung zeigt ein Beispiel für eine Berichtseite und die möglichen Arten von Verweisen.

¹⁶² Bylaws of Trusted Computing Group, https://www.trustedcomputinggroup.org/about/tcg_bylaws.pdf (letzter Abruf 23. September 2004).

¹⁶³ http://www.lfu.baden-wuerttemberg.de/lfu/uis/globus_direkt/globus4/7-fzk/gl4-5.2.html (letzter Abruf 2. Oktober 2004).

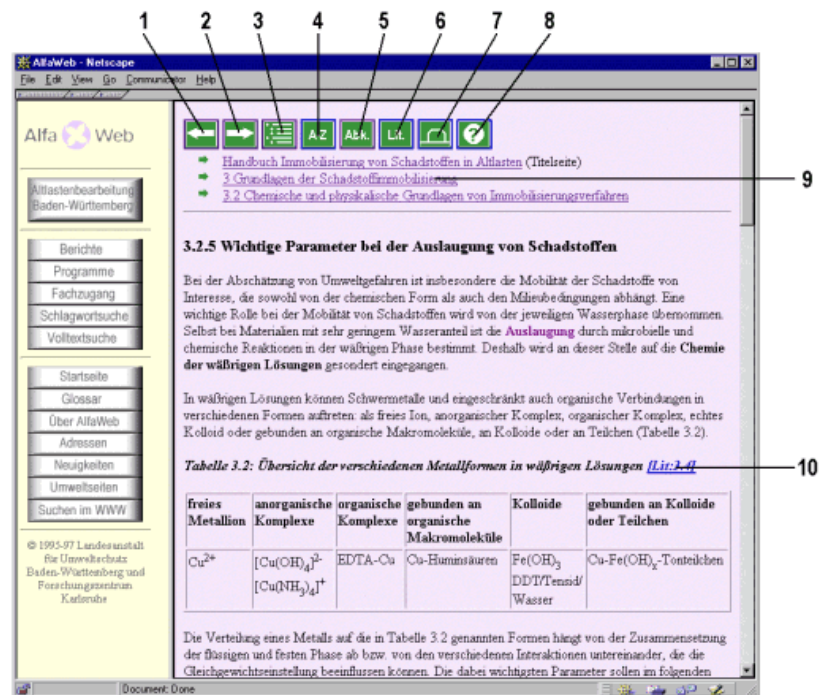


Abbildung: Beispiel für die Vernetzung von Berichtsseiten:

- | | |
|---------------------------------------|--|
| 1 = Verweis zur vorhergehenden Seite | 7 = Lesezeichen setzen |
| 2 = Verweis zur nachfolgenden Seite | 8 = Hilfe |
| 3 = Verweis zum Inhaltsverzeichnis | 9 = Verweise zum Anfang übergeordneter Kapitel |
| 4 = Verweis zum Indexverzeichnis | 10 = Verweis auf eine andere Berichtsseite |
| 5 = Verweis zum Abkürzungsverzeichnis | |
| 6 = Verweis zum Literaturverzeichnis | |

Das **Konvertierungsprogramm** ist eine Software die zwischen einem Betriebssystem und einer Applikationssoftware Interoperabilität, d.h. Kommunikationsfähigkeit herstellt.

8.2.6.8 Embedded Systems

Es handelt sich i.d.R. um **Ein-Chip-Computersysteme**, die in Produkte integriert werden und deren Funktionalitäten steuern. Embedded Systeme bestehen u.a. aus reduzierten Real-time-Betriebssystemen sowie Kommunikations- und Anwendungssoftware. Anwendungsfälle reichen vom Kaffeevollautomaten, über Kraftfahrzeuge bis hin zu Telekommunikationsendgeräten. Derartig, in einen Chip integrierte Software, weist den für die Patentfähigkeit notwendigen technischen Charakter auf, da durch sie technische Geräte in ihren Funktionalitäten gesteuert werden.

8.2.6.9 Patentfähigkeit von Schnittstellen „als solchen“

„Als solche“ kann Software und damit auch eine Softwareschnittstelle zwar wegen des Ausschlusses von der Patentfähigkeit durch § 1 Abs. 2 Nr. 3 PatG nicht Gegenstand eines Patentanspruches sein, ist sie jedoch Teil eines ansonsten patentfähigen Erzeugnisses oder Verfah-

rens, kann sie am Patentschutz teilhaben (§ 1 Abs. 3 PatG)¹⁶⁴. Der Grad der Monopolwirkung die von Patenten ausgeht, die computerimplementierte Erfindungen beinhalten, hängt somit auch von der Genehmigungspraxis der Patentämter ab. Wie die Untersuchung u.a. der Patenterteilungspraxis des EPA gezeigt hat, kann eine „liberale“ Patenterteilungspraxis, aufgrund der patentrechtlichen Sperrwirkung, im sog. „worst-case“ zu einem stark durch Monopole geprägten Softwaremarkt führen. Hinzu kommt die Gefahr mangelnder Überschaubarkeit der Schutzwirkungen computerimplementierter Erfindungen, die auf dem Gebiet der Informationstechnik ungleich höher ist, als bei Erfindungen auf den Gebieten klassischer Ingenieurdisziplinen¹⁶⁵. Vor diesem Hintergrund muss daher die weitere Entwicklung der Patentierungspraxis des EPA mit Aufmerksamkeit beobachtet werden. Setzt sich der aktuell zu beobachtende Trend betreffend der Patentierung von computerimplementierten Erfindungen fort, sind wegen der Struktur der Softwaremärkte in Deutschland erhebliche, die Marktentwicklung behindernde Rückwirkungen auf die Softwarebranche in dem v.g. Sinne zu erwarten (vgl. Ausführungen zum Softwaremarkt in 3).

Wenn auch die Softwareschnittstelle in der Gesamtsicht nur einen Teil des geschützten Erzeugnisses oder Verfahrens darstellt, nimmt sie insgesamt an der Schutzwirkung des Patents teil. Da eine Patentierung von Software „als solche“ sowohl nach deutschem PatG¹⁶⁶, als auch nach den Regelungen des EPÜ, wie aufgezeigt, ausgeschlossen ist, stellt sich der Frage eines **isolierten Patentschutzes für Softwareschnittstellen** nicht. Aus diesem Grund wurden bislang in Europa offenbar auch keine Versuche unternommen, Softwareschnittstellen als solche patentieren zu lassen. Entscheidend ist somit die patentrechtliche Beurteilung der Durchführung von Maßnahmen des Reverse Engineering.

8.2.7 Zwischenergebnis: Patentschutz von Softwaresystemen

In allen dargestellten Anwendungsfällen von Softwaresystemen besteht somit grundsätzlich im Einzelfall die Möglichkeit des Patentschutzes, so dass die Frage entsteht, ob und unter welchen Voraussetzungen in diesen Fällen Interoperabilität hergestellt werden kann.

8.3 Wirkung ausländischer Patente - Territorialitätsgrundsatz

Schutzrechte für Erfindungen kann jeder Staat nach dem „Territorialitätsgrundsatz“ nur für den räumlichen Bereich gewähren, auf den sich seine Souveränität erstreckt¹⁶⁷. Will der Berechtigte seine Erfindung auch in anderen Ländern ausschließlich, z.B. unter dem Schutz des Patentrechts, verwerten, muss er nach den dort geltenden gesetzlichen Regelungen ein entsprechendes Recht erlangen. Ein Inhaber eines U.S.-amerikanischen Patents kann daher eine Benutzung des patentierten Gegenstandes in Deutschland nicht verbieten¹⁶⁸. In derartigen Fällen wird daher die Ermittlung von Auslandsrecht insbesondere dann relevant, wenn es sich um einen

¹⁶⁴ Auf der Grundlage des EPÜ gelangt man, da die einschlägigen Regelungen inhaltsgleich sind, zu identischen Ergebnissen. Zur Patentfähigkeit von Erfindungen, die technische und nicht technische Merkmale enthalten vgl. *BGHZ* 117, 144 (148f.) - Tachcomputer. Dabei ist zu berücksichtigen, dass der in der Patentanmeldung definierte Gegenstand des Patents im Rahmen einer wertenden (Gesamt-) Betrachtung zu prüfen ist (*BGHZ* 143, 255 (263) – Logikverifikation; *BGH* CR 2004, 648 (649f.) - elektronischer Zahlungsverkehr).

¹⁶⁵ Diese These vertritt z.B. *Ensthaler*, Gewerblicher Rechtsschutz und Urheberrecht, S. 103.

¹⁶⁶ § 1 Abs. 2 Nr. 3 PatG, Art. 52 Abs. 2 lit. c EPÜ.

¹⁶⁷ *Kraßer*, Patentrecht, S. 78; *Mestmäcker/Schweitzer*, Europäisches Wettbewerbsrecht, 2. Aufl. § 26 Rdnr. 3f.

¹⁶⁸ So kommt § 139 PatG in einem solchen Fall nicht zur Anwendung; vgl. *Keuckenschrijver* in: Busse Patentgesetz § 139 Rdnr. 10 m.w.Nw.

Anwendungsfall des Art. 64 Abs. 3 EPÜ handelt, da sich die Frage der Verletzung eines Patentes bzw. die Feststellung des Umfangs von Benutzungsrechten nach dem jeweils anwendbaren nationalen Recht beurteilt. Welches nationale Patentrecht im Einzelfall zur Anwendung gelangt, ist nach international-privatrechtlichen Grundsätzen zu entscheiden. Dabei wird die vertragslose unberechtigte Benutzung eines Patents nach deliktsrechtlichen (Handlungs- oder Erfolgsort, Artt. 40, 41 EGBGB) und solche bei denen ein Lizenzvertrag besteht, nach vertragsrechtlichen Grundsätzen angeknüpft.

Das deutsche Internationale Privatrecht enthält keine einheitlichen Anknüpfungsregeln für Immaterialgüterrechte. Allgemein gilt das Schutzlandsprinzip, d.h. Entstehen, Inhalt und Erlöschen von Immaterialgüterrechten wird nach dem Recht des Staates beurteilt, in dem eine Benutzungs- oder Verletzungshandlung vorgenommen wurde¹⁶⁹. Anwendbar ist danach auch der Grundsatz der freien Rechtswahl des Art. 27 EGBGB. Dieser findet im Bereich der Immaterialgüterrecht jedoch nicht in uneingeschränktem Umfang Anwendung. Zwar ist das Schutzlandsprinzip im Fall einer Rechtswahl nicht anwendbar, jedoch gilt dies nicht für die Beantwortung der Frage von Bestand und Wirkungen des Schutzrechtes. In diesen Fällen setzt sich das Schutzlandsprinzip durch. Allerdings ist zu beachten, dass das EPÜ einheitsrechtliche Regelungen enthält¹⁷⁰. Ein U.S.-amerikanisches Patent ist somit in Europa und Deutschland nicht durchsetzbar, es sei den, im Einzelfall verweist das anwendbare IPR auf U.S.-amerikanisches Patentrecht.

Vor diesem Hintergrund wird die rechtsvereinheitlichende Bedeutung des Entwurfes der Richtlinie über patentierbare computerimplementierte Erfindungen deutlich¹⁷¹.

Zur Vermeidung der Entstehung von Schutzlücken stehen ihm folgende Handlungsalternativen zur Verfügung:

- nationale Auslandsanmeldung
- europäische Patentanmeldung (EPÜ) oder
- internationale Patentanmeldung (PCT).

Vor diesem Hintergrund ist der anwachsende Zahl von internationalen Anmeldungen beim DPMA zu werten. 84.096 PCT-Anmeldungen im ersten, internationalen Phase, des PCT-Verfahrens und 7.580 in der zweiten, nationalen Phase hatte das DPMA allein 2003 zu verzeichnen. Gegenüber dem Vorjahr bedeutete dies eine Steigerung von 16%¹⁷². Im Zuständigkeitsbereich des EPA zeigte sich 2003 eine ähnliche Entwicklung¹⁷³:

	Anmeldungen	Wachstum in %
Europäische Direktanmeldungen	55.125	2,6
Eingereichte Euro-PCT-Anmeldung (intern. Phase)	107.083	0,4
Europäische Patentanmeldungen insg. (mit Euro-PCT intern. Phase)	162.208	1,1
In regionale Phase eintretende Euro-PCT Anmeldungen	61.488	16,9
Europäische Patentanmeldungen (mit Euro-PCT regionale Phase)	116.613	9,7

¹⁶⁹ *Keuckenschrijver* in: Busse Patentgesetz Einl. Rndnr. 76 m.w.Nw.

¹⁷⁰ *Keuckenschrijver* in: Busse Patentgesetz Einl. Rndnr. 77 m.w.Nw.

¹⁷¹ Vorschlag für eine Richtlinie des Europäisches Parlaments und des Rates über die Patentierbarkeit computerimplementierter Erfindungen, KOM (2002) 92 endg.

¹⁷² DPMA, Geschäftsbericht 2003, S. 7.

¹⁷³ EPA, Geschäftsbericht 2003, S.12.

Aus diesen Entwicklungen lässt sich ableiten, dass die Anmelder die Möglichkeit eines räumlich möglichst umfassenden Schutzes anstreben. Da aber die in dem jeweiligen Schutzland bestehenden gesetzlichen Voraussetzungen erfüllt sein müssen, führt dies nicht in jedem Fall zur Erteilung eines Schutzrechtes, auch wenn dieses etwa im Heimatland des Anmelders erteilt wurde. Bedeutung erlangt ein ausländisches Schutzrecht somit nur dann, wenn ein Gegenstand in den Schutzlandstaat eingeführt wird.

9 Interoperabilität und Patentrecht

Im technischen Teils des Gutachten wurde erläutert, dass die untersuchten Softwaresysteme ohne Ausnahme, wenn auch in ggf. unterschiedlicher Intensität¹⁷⁴ auf die Herstellung von Interoperabilität angewiesen sind¹⁷⁵. Damit Interoperabilität hergestellt werden kann, müssen von den Schnittstellen „computerimplementierter Erfindungen“ gute Beschreibungen, der Quellcode oder sog. „SDK's“ zur Verfügung gestellt werden¹⁷⁶. Liegen derartige Informationen nicht vor oder sind sie qualitativ nicht ausreichend, verbleibt nur die Möglichkeit der Herstellung von Interoperabilität durch Methoden des Reverse Engineering¹⁷⁷.

Zu beantworten ist daher die Frage, inwieweit das Patentrecht Möglichkeiten eröffnet, entsprechende Informationen erhalten zu können. Die Untersuchung konzentriert sich im Folgenden, auf die interoperabilitätsrelevanten Informationsquellen Quellcode sowie Reverse Engineering.

9.1 Wirkung des Patentschutzes

Da im Patenterteilungsverfahren der Antragsgrundsatz gilt, bestimmt sich der Umfang eines erteilten Patents nach dem Inhalt des gestellten Antrags auf Patenterteilung. Es ist somit Sache des Anmelders, das Schutzbegehren zu formulieren und damit dem Schutzbereich des Patents zu bestimmen¹⁷⁸. Der Anmelder muss in der Anmeldung angeben, was als patentfähig unter Schutz gestellt werden soll (§ 34 Abs. 3 PatG). Mit Erlass des Patenterteilungsbeschlusses, der nach deutschem Patentrecht als Verwaltungsakt zu qualifizieren ist¹⁷⁹, entsteht das Schutzrecht. Nach § 49 Abs. 1 PatG erlässt die Prüfungsstelle einen entsprechenden Erteilungsbeschluss, wenn die Anmeldung den Anforderungen der §§ 34, 37 und 38 PatG genügt und der Gegenstand der Patentanmeldung nach §§ 1 bis 5 PatG patentfähig ist (vgl. Art. 97 Abs. 2 E-PÜ). Dem Patentschutz unterliegen dann **alle** der Erfindung zugehörigen Elemente, die für das Erzielen einer bestimmten Wirkung oder Herstellung eines Erzeugnisses erforderlich sind, soweit diese im Rahmen der Patentanmeldung **offenbart** wurden¹⁸⁰.

9.2 Offenlegung - Quellcode im Patenterteilungsverfahren

Auch die Eröffnung der Möglichkeit des Zugriffs auf den Quellcode ist geeignet, Interoperabilität herzustellen, da die Einpassung von Anwendungen wesentlich erleichtert wird. Beispiel hierfür ist das **Government Security Programm** (GSP) der Fa. Microsoft. Im Rahmen dieser Initiative

¹⁷⁴ Siehe Übersicht bei 5.8.

¹⁷⁵ Vgl. 5.1.2; 5.2.2; 5.3.2; 5.4.2; 5.5.2; 5.6.2; 5,7,2

¹⁷⁶ Siehe 4.3.

¹⁷⁷ Siehe 7.

¹⁷⁸ Siehe auch inhaltsgleichen Art. 69 Abs. 1 EPÜ für das europäischen Patentanmeldeverfahren.

¹⁷⁹ *Keuckenschrijver* in: Busse Patentgesetz vor § 34 Rdnr. 85 m.w.Nw.

¹⁸⁰ *Kraßer*, Patentrecht, S. 562 mit zahlreichen Rechtsprechungsnachweisen.

wird den teilnehmenden Regierungen teilweise der Einblick in den Quellcode sowie den dazugehörigen internen Microsoft-Dokumentationen von Applikationssoftware, wie z.B. „Microsoft Office 2003“ gewährt¹⁸¹. Entwicklern soll dadurch die Möglichkeit verschafft werden beispielhaft verfolgen zu können, wie das Zusammenspiel des betreffenden Betriebssystems mit anderen Diensten oder Office-Programmen funktioniert¹⁸². Es handelt sich somit um ein als „klassisch“ zu bezeichnendes Szenario, welches die Herstellung von Interoperabilität zum Ziel hat und in etwa mit der Bereitstellung von SDK's verglichen werden kann.

Fraglich ist, wie diese Handlungsoption sich in das Patentrechtssystem einfügt. Ansatzpunkt finden sich z.B. in den Prüfungsrichtlinien von DPMA und EPA bezüglich der Offenlegung der Erfindung im Anmeldeverfahren. Es besteht die Möglichkeit bei der Darstellung der Anmeldung, neben Fachausdrücken aus dem Gebiet der Datenverarbeitung, auch Strukturangaben bzw. wirkungs- und funktionsbezogene Angaben zu verwenden. Ferner ist die Verwendung von Datenfluss- und Programmablaufplänen zugelassen ebenso wie kurze Auszüge aus einem Programm für DV-Anlagen in einer üblichen, genau bezeichneten Programmiersprache¹⁸³.

In der Anmeldungspraxis wird auf diese Möglichkeit offenbar im Regelfall nicht zurückgegriffen, da Merkmale, die nicht als erfinderisch gelten, auch nicht offenbart werden müssen¹⁸⁴. Eine generelle Offenlegungspflicht könnte sich, zunächst nur dann durchgesetzt, wenn es zu einem originären Patentschutz von Computerprogrammen kommen würde¹⁸⁵.

Wenn im Rahmen des GSP-Programms die zumindest teilweise Offenlegung des Quellcodes als gangbarer Weg zur Herstellung von Interoperabilität betrachtet wird, kann diese Folgerung auch auf das Patentanmeldeverfahren grundsätzlich übertragen werden. Allerdings muss sichergestellt sein, dass die offen gelegten Informationen in entsprechender Qualität vorgelegt werden. Berücksichtigt man die hohe Belastung der Prüfungsstellen, mögen Zweifel angebracht sein, da die Tauglichkeit sich letztlich erst im Rahmen der Programmierung interoperabler Software herausstellen wird¹⁸⁶.

Da in der Patentanmeldepraxis von der Möglichkeit der Hinterlegung von Quellcodes nicht in nennenswertem Umfang Gebrauch gemacht wird, ist der Herstellung von Interoperabilität mittels Zugriff auf den Quellcode (der Schnittstellen) nicht sichergestellt¹⁸⁷.

9.3 Benutzung i.S.d. § 9 PatG

§ 9 PatG legt fest, welche Wirkungen ein Patent entfaltet. Danach ist allein der Patentinhaber zur Nutzung der Erfindung befugt. Ihm wird somit ein alleiniges Benutzungsrecht (sog. zeitlich begrenztes Monopol) und ein umfassendes Verwertungsrecht zugewiesen¹⁸⁸.

Fraglich ist, welche Konsequenzen sich aus der Tatsache ergeben, dass, aus Gründen der Herstellung von Interoperabilität, im Fall der Benutzung einer Schnittstelle u.U. lediglich ein Teil

¹⁸¹ www.heise.de/newsticker/meldung/51244.

¹⁸² www.heise.de/newsticker/meldung/31394.

¹⁸³ DPMA, Richtlinien für die Prüfung von Patentanmeldungen, Ziff. 4.3.7 (Stand 1. März 2004); EPA, Richtlinien für die Prüfung im Europäischen Patentamt Kapitel IV, Teil C Kap. II Nr. 4.14a Computerprogramme (Stand Dezember 2003).

¹⁸⁴ *Kiesewetter-Köbinger*, GRUR 2001, 185 (190).

¹⁸⁵ Vgl. hierzu *Weyand/Haase*, GRUR 2004, 198 (203).

¹⁸⁶ Zur Problematik der Qualität hinterlegter Quellcodes; *Hoeren*, CR 2004, 721ff.

¹⁸⁷ Vgl. hierau auch *Tauchert*, GRUR 2004, 922 (923).

¹⁸⁸ *Keuckenschrijver* in: Busse Patentgesetz § 9 Rndnr. 43 m.w.Nw.

des geschützten Erzeugnisses oder Verfahrens verwendet wird¹⁸⁹. Im Fall des Erzeugnispatents liegt nach gefestigter Rechtsprechung eine Benutzungshandlung nur dann vor, wenn von der Gesamtheit der Merkmale Gebrauch gemacht wird¹⁹⁰. Werden nur Teile eines Erzeugnisses hergestellt, soll nur bei durch den Benutzer intendierter Umgehungsabsicht eine Patentverletzung vorliegen¹⁹¹. Im Fall eines Verfahrenspatents wird ein bestimmter Verfahrensablauf geschützt. Eine das Verfahrenspatent verletzende Anwendung (§ 9 Nr. 2 PatG) liegt folglich dann vor, wenn der Dritte sich des geschützten Verfahrensablaufs bedient.

Wie im technischen Teile des Gutachtens (6.3) dargestellt wurde, kommt im Rahmen der Herstellung von Interoperabilität dem Einsatz von Methoden des Reverse Engineering besondere Bedeutung zu. Um die notwendigen Schnittstelleninformationen erhalten zu können, muss, wie dort aufgezeigt, das Erzeugnis oder Verfahren als Ganzes analysiert werden. Mithin wird in diesen Fallkonstellationen notwendiger Weise nicht nur ein Teil (Schnittstelle) des dem Patentschutz unterliegenden Gegenstandes oder Verfahrens benutzt. Die vorgehend beschriebenen Einschränkungen bei lediglich teilweiser Benutzung eines Patents greifen in dem hier zu beurteilenden Fall somit nicht.

Wenn auch die Softwareschnittstelle in der Gesamtsicht nur einen Teil des geschützten Erzeugnisses oder Verfahrens darstellt, nimmt sie insgesamt an der Schutzwirkung des Patents teil. Da eine Patentierung von Software „als solche“ sowohl nach deutschem PatG¹⁹², als auch nach den Regelungen des EPÜ, wie aufgezeigt, ausgeschlossen ist, stellt sich der Frage eines **isolierten Patentschutzes für Softwareschnittstellen** nicht. Aus diesem Grund wurden bislang in Europa offenbar auch keine Versuche unternommen, Softwareschnittstellen als solche patentieren zu lassen. Entscheidend ist somit die patentrechtliche Beurteilung der Durchführung von Maßnahmen des Reverse Engineering.

9.4 Interoperabilität durch Reverse Engineering

Exemplarisch soll an dieser Stelle erläutert werden, wie sich in groben Zügen die Vorgehensweise beim Reverse Engineering von Softwareschnittstellen darstellt:

Ein Softwarehersteller erhält eine Anfrage von einem Kunden, ob dieser eine Applikationssoftware, die individuelle auf die bei dem Kunden bestehenden besonderen Bedürfnisse angepasst ist, erstellen kann. Der Hersteller des Systems hat die Softwareschnittstelle und das Datenformat nicht offen gelegt, auf die die Applikationssoftware zugreifen muss. Daher analysiert der Softwareersteller der Applikationssoftware den Output und den Softwareteil, der für die Aufbereitung des Outputs verantwortlich ist, um auf diesem Wege Struktur und Verfahren der Kommunikation kennen zu lernen und leitet hieraus ab, in welcher Weise auf den Output des Systems zugegriffen werden kann.

Die dargestellte Patenterteilungspraxis zeigt, dass Softwareschnittstellen als unselbständiger Teil einer computerimplementierten Erfindung Patentschutz erlangen können, sofern die Software mit einem patentfähigen Erzeugnis oder Verfahren verbunden wird. Handelt es sich in dem vorgehend beschriebenen Fall bei dem Betriebssystem um eine computerimplementierte Erfindung, würde sich der Dritte im Rahmen der Analyse des Dokumentes der unter Paten-

¹⁸⁹ Im Fall der Verletzung eines Europäischen Patents ist nach Art. 64 Abs. 3 i.V.m. Abs. 1 EPÜ das Recht des Vertragsstaates heranzuziehen, in dem das Patent erteilt wurde. Für in Deutschland erteilte Europäischen Patente sind somit die Regelungen des deutschen PatG maßgeblich. Für den Bereich des Urheberrechts instruktiv *BGH GRUR* 2003, 786ff. – Innungsprogramm.

¹⁹⁰ *Keuckenschrijver* in: Busse Patentgesetz § 9 Rndnr. 65 m.w.Nw.

¹⁹¹ Vgl. *BGHZ* 2, 387 (391) – Mülltonne.

¹⁹² § 1 Abs. 2 Nr. 3 PatG, Art. 52 Abs. 2 lit. c EPÜ.

schutz stehenden Kommunikationsregeln der Softwareschnittstelle bedienen. Fraglich ist, ob diese Vorgehensweise einer Benutzung gleichkommt und ob diese das Patent verletzt¹⁹³.

9.4.1 Begriff des Reverse Engineering

Eine klassische Definition des Begriffs Reverse Engineering findet sich unter der Internetadresse http://de.wikipedia.org/wiki/Reverse_engineering. Dort wird der Begriff wie folgt definiert:

*„**Reverse Engineering** bezeichnet den Vorgang, aus einem bestehenden fertigen System oder einem meist industriell gefertigten Produkt, durch Untersuchung der Strukturen, Zustände und Verhaltensweisen die Konstruktionselemente zu extrahieren. Aus dem fertigen Objekt wird somit wieder ein Plan gemacht. Im Gegensatz zu einer funktionellen Nachempfindung, die ebenso auf Analysen nach dem Black Box Prinzip aufbauen kann, versucht das Reverse Engineering das vorliegende Objekt weitgehend exakt abzubilden. Es wird somit ermöglicht, eine 1:1 Kopie des Objekts zu erstellen und auf deren Basis Entwicklung zu betreiben. Speziell bezogen auf Computersoftware wird darunter meistens einer der beiden folgenden Vorgänge verstanden:*

- *Die Rückgewinnung des Quellcodes oder einer vergleichbaren Beschreibung aus Binärcode, z. B. von einem ausführbaren Programm oder einer Programmbibliothek oder*
- *die Erschließung der Regeln eines Kommunikationsprotokolls aus der Beobachtung der Kommunikation.*

Im ersten Fall werden oft Decompiler eingesetzt, die den Quellcode eines Programms weitestgehend automatisch aus seinem Binärcode zurückgewinnen. Ist dies nicht durchgehend möglich, so kann auch der aus dem Binärcode des Programms direkt ermittelbare Maschinencode manuell analysiert werden, was allerdings eine merkliche Erschwernis bedeutet. Es kann i.d.R. nicht der komplette Programmquellcode ermittelt werden, da z. B. Kommentare und lokale Objektname nur selten im verfügbaren Binärcode enthalten sind. Oft ist das Ergebnis jedoch trotzdem für den jeweiligen Zweck ausreichend, z. B. zur Verhaltensanalyse eines Softwaresystems oder als Hilfe beim Beheben eines Fehlers.“

Dieser Definitionsversuch vermag die tatsächlichen Abläufe im Zusammenhang mit der Vorname eines Reverse Engineering einer Softwareschnittstelle nicht hinreichend exakt genug zu beschreiben. So dürfte es nur in Ausnahmefällen tatsächlich zu der Erstellung einer sog. „1:1 Kopie“ der analysierten Softwareschnittstelle kommen. Ziel der **Dekompilierung** ist vielmehr lediglich die Gewinnung von Strukturinformationen. Die reale programmtechnische Umsetzung wird sich dagegen in aller Regel nicht an der Vorlage ausrichten und schon gar nicht wird der aufwendige Versuch der Herstellung einer 1:1 Kopie unternommen werden. Denn es wird nicht die Software selbst kopiert, sondern nur die Kommunikationsfähigkeit einer Software mit der Schnittstelle hergestellt.

An dieser Stelle kommt ein weiterer wesentlicher Nachteil des Reverse Engineering zum Tragen. Denn anders als bei chemischen oder pharmazeutischen Stoffen, kann ein Reverse Engineering einer Software nicht in jedem Fall zu derart validen Ergebnissen führen, die ein Nachprogrammieren ermöglichen. **Chemische und pharmazeutische Analyseverfahren** führen demgegenüber zu Ergebnissen, die die Zusammensetzung des untersuchten Stoffes exakt feststellen können. Im Bereich der Softwareanalyse ist dies dagegen nicht möglich. Mittels des Reverse Engineering erlangt man nicht den **Sourcecode**, der den Dritten in die Lage versetzen würde, eine Software exakt nachzuprogrammieren. Ermittelt werden kann nur der sog. **Assemblercode**, der im Übrigen durch große Unübersichtlichkeit gekennzeichnet ist¹⁹⁴. Folge dieser Tatsache ist, dass die Analyse einer Software, die den Problemlösungsansatz der untersuchten Ausgangssoftware offen zulegen vermag, äußert fehlerbehaftet und zeitaufwendig ist.

¹⁹³ Das EPÜ sieht keine eigenständigen Vorschriften über die Verletzung eines Europäischen Patents vor und verweist dementsprechend durch Art. 64 Abs. 3 EPÜ auf das anwendbare nationale Recht.

¹⁹⁴ Vgl. *Siepmann*, Softwarepatente/ Open Source – Öffentliches Expertengespräch des Ausschusses für Kultur und Medien sowie des Rechtsausschusses des Deutschen Bundestages 2001, S. 11 (vgl. http://www.bundestag.de/gremien/welt/glob_end/v12_literatur_lm.html letzter Abruf 2. Oktober 2004).

Der Dritte ist daher z.B. im Rahmen der Programmierung einer Applikationssoftware dazu gezwungen im Wege eines Trial-and-Error-Verfahrens die Kommunikationsfähigkeit der Software schrittweise herzustellen. Solange das Verhältnis zwischen einer zeitaufwendigen und mit hoher Fehlerwahrscheinlichkeit behafteten Dekompilierung zu dem erzielbaren Umsatz bzw. der Erlangung von Marktanteilen in einem wirtschaftlichen Verhältnis zueinander steht, wird ein softwareherstellender Dritter, trotz der genannten Nachteile, ein Reverse Engineering durchführen.

Einsatzfelder des Reverse Engineering im Bereich Software sind z.B.:

- Software Maintenance (Fehlerbeseitigung und Anpassung der Software neue Anforderungen)
- Evaluation
- Herstellung von Interoperabilität
- RE-Engineering (Ersatzprodukte)
- Wissenschaftliche Forschung
- Verbesserung der Sicherheit
- Verbesserung der Performance

Trotz der Einschränkungen hinsichtlich der zu erwartenden Qualität ist das Reverse Engineering ein gutes und in der Praxis intensiv genutztes Instrument zur Herstellung von Interoperabilität. Im Rahmen der Diskussion der patentrechtlichen Zulässigkeit der Durchführung von Maßnahmen des Reverse Engineering ist die unterschiedliche Zielrichtung, mit der ein Reverse Engineering durchgeführt werden kann zu berücksichtigen. In diesem Kurzgutachten wird nur der Aspekt der Herstellung von Interoperabilität angesprochen.

9.4.2 Reverse Engineering im Patentrecht

Sucht man nach einer Regelung im Patentrecht, die sich auf einen zumindest ähnlichen Lebenssachverhalt bezieht zeigt sich, dass, neben der Regelung in § 69e UrhG, nur im Halbleiterschutzgesetz eine ausdrückliche Regelung in § 6 Abs. 2 Nr. 2 und 3 enthalten ist, die eine das Reverse Engineering ansprechende Situation erfasst. Danach ist ein Reverse Engineering nur dann erlaubt, wenn es im privaten Bereich zu nichtgeschäftlichen Zwecken erfolgt, es dem Zweck der Analyse, Bewertung oder Ausbildung dient und es sich nicht um eine reine „Nachbildung“ handelt. Ob diese Grundsätze ohne weiteres auf das PatG bzw. GebrMG übertragen werden können ist äußerst fraglich, denn das HlSchG hat lediglich gegenüber dem PatG lediglich Ergänzungsfunktion in Bezug auf Gegenstände, die dem Patentschutz zwar ihrer Art nach zugänglich, aber nicht schutzwürdig sind sowie dessen Ausgestaltung als sog. Leistungsschutzrecht. Erfindung auf der einen und Leistung auf der anderen Seite, sind wesensverschieden und klar voneinander zu trennen. Leistungsschutzrechte sind daher enumerativ ausgestaltet, weshalb eine entsprechende Anwendung der Regelungen des HlSchG auf patentrechtliche Fallgestaltungen in dem hier angesprochenen speziellen Fall ausscheiden¹⁹⁵.

Regelungen im Patentrecht, die Fallgestaltungen des Reverse Engineering zu erfassen vermögen, können §§ 9 und 11 PatG entnommen werden.

¹⁹⁵ Zum Verhältnis von originären Schutzrechten und Leistungsschutzrechten im Urheberrecht *Rehbinder*, Urheberrecht, Rdnr. 394.

9.4.2.1 Zulässige Maßnahmen im Rahmen der § 9 PatG

In § 9 PatG werden die Wirkungen des Patents geregelt, d.h. die Reichweite der Benutzungsrechte von Lizenznehmern sind auf der Grundlage dieser Regelung zu bestimmen, soweit keine anderweitige vertraglich Regelung getroffen wurde. Ob die Analyse von Softwareschnittstellen, die Teil einer computerimplementierten Erfindung zur Herstellung von Interoperabilität als zulässige Benutzung nach § 9 PatG qualifiziert werden kann, ist bis dato noch nicht ersichtlich Gegenstand gerichtlicher Entscheidungen gewesen. Wendet man allgemeine, § 9 PatG zugrunde liegende Maßstäbe an, ergibt sich folgendes Bild:

Maßnahmen der **Ausbesserung** und der Betriebsstofferneuerung stellen keinen Patenteingriff dar. Hierunter werden Inbetriebnahme, Inbetriebhaltung und Pflege verstanden¹⁹⁶. Entscheidend ist in diesen Fällen, dass die Handlung nicht der Neuherstellung des Erzeugnisses gleichkommt¹⁹⁷. Eine Verwirklichung der Erfindung wird auf in dem Umbau eines Erzeugnisses gesehen und damit der Neuherstellung gleichgestellt¹⁹⁸. Auf Verfahrenspatente ist diese Regelungssystematik gleichfalls anzuwenden¹⁹⁹. Das PatG sieht somit zwar kein **Reparaturmonopol** des Patentinhabers vor²⁰⁰, allerdings werden die in § 9 PatG enthaltenen Verbotstatbestände als erfüllt angesehen, wenn

- selbständig geschützte Teile einer Vorrichtung **erneuert** werden,
- eine Rekonstruktion des Erzeugnisses aus Teilen unbrauchbar gewordener Erzeugnisse erfolgt sowie
- Recycling, wenn die Erzeugniseigenschaften verloren gegangen waren²⁰¹.

Diese Handlungen werden der einer **Neuherstellung** gleichartig angesehen.

Die entscheidende Frage ist an dieser Stelle die, ob bei einem Erzeugnispatent in der Dekompilierung, oder dem Durchführen von Maßnahmen des Reverse Engineering eine bestimmungsgemäße Benutzung nach § 9 Satz 2 Nr. 1 PatG gesehen werden kann. Werden wesentliche Elemente einer patentierten computerimplementierten Erfindung dagegen in einen anderen Programmzusammenhang gestellt, also die Schnittstelle benutzt wird um Applikationssoftware Dritter mit einer Betriebssystemsoftware interoperabel zu machen, wird hierin eine dem Patentinhaber vorbehaltene Herstellung gesehen²⁰². Im Bereich chemischer Produkte gibt es der Verwendung von Schnittstellenkommunikationsfunktionen in etwa ähnlich gelagerte Fallgestaltungen. Wird ein patentierter chemischer Stoff umgesetzt und nimmt das Verarbeitungsprodukt vorteilhafte Eigenschaften und Wirkungen des Ausgangsstoffes an, die in erheblichem Maß durch diesen geprägt werden, wird eine das Patent verletzende mittelbare Benutzung bejaht²⁰³. Da der Kommunikationsfähigkeit entscheidende Bedeutung zukommt, liegt der Schluss nahe,

¹⁹⁶ *Keuckenschrijver* in: Busse Patentgesetz § 9 Rdnr. 70.

¹⁹⁷ *Keuckenschrijver* in: Busse Patentgesetz § 9 Rdnr. 70 m.w.Nw.

¹⁹⁸ *Keuckenschrijver* in: Busse Patentgesetz § 9 Rdnr. 69.

¹⁹⁹ *Keuckenschrijver* in: Busse Patentgesetz § 9 Rdnr. 83.

²⁰⁰ *BGHZ 2, 387 – Mülltonne*; zum Thema Patentschutz Reparatur- und Ersatzteile, vgl. *Rübel*, GRUR 2002, 561ff.

²⁰¹ *Keuckenschrijver* in: Busse Patentgesetz § 9 Rdnr. 70 m.w.Nw.

²⁰² *v. Falck/Plassmann* in: Kilian/Heussen ComputerrechtsHandbuch, Kap. 52 Rdnr. 81 unter Hinweis auf BGH GRUR 1973, 518 (522) – Spielautomat.

²⁰³ *Kraßer*, Patentrecht, S. 778 m.w.N..

dass aus prägende Eigenschaften und Wirkungen des Ausgangserzeugnisses oder –verfahrens zurückgegriffen wird²⁰⁴.

Handelt es sich um eine Verfahrenspatent stellt sich die Frage, ob es sich bei der Dekompilierung der Software um zulässige Vorbereitungshandlungen für den Gebrauch des geschützten Verfahrens, wie z.B. das Herrichten von bei der Benutzung des Verfahrens benötigter Gegenstände bzw. Mittel nach § 9 Satz 2 Nr. 2 PatG oder eine rechtsverletzende Anwendung des Verfahrenspatents handelt²⁰⁵. Das Reverse Engineering ist zunächst nicht darauf gerichtet, ein unmittelbares Verfahrenserzeugnis zu erstellen. Sondern mittels Rückgewinnung des Quellcodes oder einer vergleichbarer Beschreibung bzw. die Erschließung der Regeln eines Kommunikationsprotokolls soll die Möglichkeit für den Dritten geschaffen werden, ein mit Verfahrensprodukten kommunikationsfähiges eigenständiges Produkt entwickeln zu können. Letztlich identisch sind daher nur die verwendeten Kommunikationsregeln. Für den Bereich der Computersoftware liegt jedoch noch keine Rechtsprechung vor, ob die Verwendungen des Verfahrens, hier der Schnittstelle, wesentlich zur Herstellung eines Erzeugnisses (Applikationssoftware des Dritten) beigetragen hat²⁰⁶.

In diesem Zusammenhang ist zu berücksichtigen, dass die Erprobung eines Verfahrenspatents, die auf den Nachweis ordnungsgemäßer Funktion gerichtet ist, als patentverletzend angesehen wird²⁰⁷. Da die Kommunikationsfähigkeit zwischen dem durch ein Patent geschützten Verfahren und dem Erzeugnis oder Verfahren eines Dritten nur dann zur vollen Funktionsfähigkeit der Anwendungen führt, kommt es entscheidend auf den Aspekt ordnungsgemäßer Funktion in dem o.g. Sinne an²⁰⁸.

Ob Maßnahmen des Reverse Engineering im Rahmen des § 9 PatG als zulässig qualifiziert werden können, ist zumindest zweifelhaft, zumal in diesem Bereich erhebliche Abgrenzungsschwierigkeiten bestehen und sich, hierdurch bedingt, in der Rechtsprechung kein einheitlicher Beurteilungsmaßstab herausbilden konnte²⁰⁹. Es verbleibt mithin hinsichtlich der Einordnung von Maßnahmen des Reverse Engineering als zulässige Benutzung i.S.d. § 9 PatG bei einem hohen Maß an Rechtsunsicherheit.

9.4.2.2 Zulässige Maßnahmen im Rahmen der § 11 PatG

Für private nicht gewerbliche Nutzungen und Handlungen zu Versuchszwecken werden nach § 11 PatG die Wirkungen des Patents begrenzt. Zwar wird dem Patentinhaber die Befugnis eingeräumt, jedem Dritten die gewerbliche Verwertung der geschützten Erfindung zu verbieten, Forschung kann er aber nicht verhindern. Versuchshandlungen stellen keine Patentverletzung dar, soweit sie der Gewinnung von Erkenntnissen und damit der **wissenschaftlichen Forschung** über den Gegenstand der Erfindung, einschließlich dessen Verwertung, dienen. Entscheidender Bedeutung kommt im Rahmen der Diskussion, die zum Anwendungsbereich des

²⁰⁴ Hier ist zu berücksichtigen, dass die Herstellung erfindungsfunktionell individualisierte Teile (Abbildung der Schnittstelle) den Tatbestand des Herstellens nach § 9 PatG dann erfüllen, wenn Einzelteile nicht ohne weiteres auch außerhalb der geschützten Gesamtvorrichtung verwendet werden können und eine solche Ausgestaltung erhalten haben, die sie durch ihre erfindungsgemäße Anpassung an die geschützte Gesamtvorrichtung aus der Zahl anderer vergleichbarer Einzelteile heraushebt und durch eine solche Individualisierung in unmittelbarer Beziehung zum Erfindungsgedanken steht (BGH GRUR 1971, 78ff. – Dia-Rähmchen).

²⁰⁵ Anders wenn ein Fall einer mittelbaren Patentverletzung vor liegt BGH GRUR 1992, 305 – Heliumeinspeisung.

²⁰⁶ v. Falck/Plassmann in: Kilian/Heussen ComputerrechtsHandbuch, Kap. 52 Rdnr. 90.

²⁰⁷ Keuckenschrijver in: Busse Patentgesetz § 9 Rdnr. 87 m.w.N.; wohl noch str.

²⁰⁸ Auch hier liegt noch keine Rechtsprechung mit Bezug zu Computerprogrammen vor.

²⁰⁹ Kraßer, Patentrecht, S. 784.

§ 11 Nr. 2 PatG in Bezug auf die Zulässigkeit des Reverse Engineering zu führen sein wird, der Frage zu, ob es sich um Versuche in dem dort angesprochenen Sinne handelt²¹⁰.

Patentschutz ist nach Auffassung des Gesetzgebers dort nicht gerechtfertigt, wo die Weiterentwicklung der Technik behindert würde (sog. **Forschungsprivileg**)²¹¹. Dient der Versuch der Weiterentwicklung des Produkts oder dessen wirtschaftlicher Verwertung, soll das Forschungsprivileg jedoch nicht greifen²¹². Vielfach wird daher in der Literatur die Auffassung vertreten, dass die Ausrichtung von Versuchen im Rahmen des § 11 Nr. 2 PatG auf die Gewinnung neuer Erkenntnisse gerichtet sein muss und dabei eine Ausrichtung auf deren objektive Eignung zu diesem Zweck gegeben sein müsse²¹³.

Ausgangspunkt ist somit die im Patentrecht enthaltene Informations- und Anspornfunktion, die die Wirkungen des Patents nicht auf Handlungen erstreckt, die Versuchszwecken dienen. Dies soll die Weiterentwicklung oder Verbesserung der geschützten Erfindung ermöglichen. Im Regelfall führt dies zur Entwicklung abhängiger Erfindungen bzw. andersartiger oder weiterführender Lösungen²¹⁴. Entscheidend ist dabei, dass die Erprobung zur Beseitigung bestehender Unsicherheiten über den Gegenstand der Erfindung (Patent) einschließlich seiner Verwendung gewonnen werden sollen²¹⁵.

Weder in der Literatur, noch in der Rechtsprechung wurde bisher die Frage erörtert, ob die vorgehend dargestellten Grundsätze auf Fallgestaltungen des Reverse Engineering im Bereich der Herstellung von Interoperabilität von Softwaresystemen übertragen werden können²¹⁶. Beim Reverse Engineering von Software geht es nicht um die Beseitigung von Unsicherheiten über die Wirkungsweise der Erfindung, deren Weiterentwicklung oder die Gewinnung neuer Erkenntnisse über die Software, sondern um die Herstellung von Interoperabilität – Austausch von Informationen -, die zur Benutzung, zumindest von Teilen des durch Patent geschützten Erzeugnisses oder Verfahrens (Schnittstelle) führt. Die von dem Dritten entwickelten Erzeugnisse oder Verfahren sollen mit den untersuchten Softwaresystemen verwoben werden um gemeinsam Aufgaben in einem Gesamtsystem lösen zu können. Für diesen Fällen ist daher zu klären, ob vielmehr allein kommerzielle Verwertungshandlungen vorgenommen werden, denen der Charakter von Versuchen kaum beigemessen werden kann²¹⁷. Da weder die „Qualitätskontrolle“ noch die Weiterentwicklung des untersuchten Erzeugnisses oder Verfahrens Zweck der Durchführung des Reverse Engineering ist, kann, unter Anwendung der Grundsätze der Entscheidung des Court of Appeal in der Entscheidung „Touchdown“ entwickelten Grundsätze der Versuchscharakter des Reverse Engineering durchaus verneint werden²¹⁸. Da die Durchführung

²¹⁰ So findet sich unter www.computerbase.de/lexikon/Patent (letzter Abruf 16. November 2004) folgender Definitionsversuch: „Versuch ist jedes planmäßige Vorgehen zur Gewinnung neuer Erkenntnisse, die sich auf die benutzte Erfindung selbst beziehen“.

²¹¹ *BGHZ* 130, 259ff.

²¹² *Keuckenschrijver* in: Busse Patentgesetz § 11 Rdnr. 16.

²¹³ *Schramm* (Hrsg.), *Der Patentverletzungsprozeß*, 4. Aufl., S. 113, unter Hinweise auf die Entscheidungen des BGH „Klinische Versuche I und II“.

²¹⁴ *Kraßer*, *Patentrecht*, S. 813.

²¹⁵ *BGHZ* 135, 217 (226f.) – *Klinische Versuche II* = *NJW* 1997, 3092 (3094).

²¹⁶ Gleiches gelte für die Frage, ob der Patentinhaber in diesen Fallgestaltungen den Schranken der Sozialbindung des Eigentums im Sinne der Entscheidung des *BVerfG* aus Gemeinwohlgründen unterworfen ist und entsprechende Eingriffe in seine Eigentum (Patent) hinnehmen muss, *BVerfG*, 1 BvR 1864/95, Online-Version Abs. 22 (www.bundesverfassungsgericht.de/entscheidungen/rk20000510_1bvr186495, letzter Abruf 15. Nov. 2004).

²¹⁷ Allgemein zur Frage, wann von Versuchen gesprochen werden kann, *Keuckenschrijver* in: Busse Patentgesetz § 11 Rdnr. 17, unter Hinweis auf *BGH NJW* 1997, 3092 (3094) – *Klinische Versuche II* und *CA GRUR Int.* 1987, 108 – *Touchdown*.

²¹⁸ *CA GRUR Int.* 1987, 108 – *Touchdown*.

des Reverse Engineering von Software zum Zweck der Herstellung von Interoperabilität ausschließlich auf die Kommunikationsfähigkeit eines eigenen Softwareprodukts gerichtet ist, liegt der Schluss nahe, dass die nicht die Erfindung selbst Gegenstand des Versuchs ist, sondern nur als Mittel eingesetzt wird, um das eigene Software-Produkt marktfähig, d.h. interoperabel, zu machen²¹⁹. Den Rahmen dieses Kurzgutachtens würde die Erörterung der Frage sprengen, inwieweit die Entwicklung (nicht Fortentwicklung) interoperabler Softwaresysteme im Interesse der Allgemeinheit geboten ist (Art. 14 GG). Es erscheint zumindest als zweifelhaft die Gemeinwohlanalyse, die das BVerfG und der BGH in den Entscheidungen „Klinische Versuche“ den Entscheidungen, Versuchshandlungen nach § 11 Nr. 2 PatG vom Patentschutz freizustellen, zugrundegelegt haben, auf den Bereich der Softwareentwicklung zu übertragen. Denn im Regelfall geht es in den hier angesprochenen Fällen nicht um faktischen Konkurrentenschutz, da Interoperabilität zwischen Softwaresystemen hergestellt wird, die unterschiedlichen Produktkategorien zuzuordnen sind (z.B. Betriebssystem/ Applikationssoftware, wie etwas CAD-Programme).

Den von *Weyand/Haase*²²⁰ geäußerten Bedenken folgend, kann nach hier vertretener Auffassung de lege lata nur zu nicht gewerblichen Zwecken für Softwareschnittstellen, die Teil einer computerimplementierte Erfindungen sind, Interoperabilität durch Reverse Engineering hergestellt werden. Die durch § 69e UrhG gesetzten Rahmenbedingungen bleiben bestehen. Bei Untersuchungen, die nicht als Versuche i.S.d. § 11 Nr. 2 PatG qualifiziert werden können, greift das Forschungsprivileg somit nicht. Dieses Ergebnis steht, dann aber wiederum in Widerspruch zu den urheberrechtlichen Ausnahmebestimmungen²²¹. Hiervon geht eine nicht zu unterschätzende Gefahr für im Rahmen der Softwareentwicklung notwendigen **Standardisierungsprozesse** aus²²².

Im Fall der Herstellung von Interoperabilität mittels Reverse Engineering von Softwareschnittstellen kommt man zu den folgenden Ergebnissen:

- Da nicht die computerimplementierte Erfindung „nachprogrammiert“ wird, sondern mittels der Softwareschnittstelle lediglich z.B. die Kommunikation zwischen Applikationssoftware und Betriebssystem hergestellt wird (Interoperabilität), liegt kein Fall der 1:1-Kopie eines Erzeugnisses vor
- Es werden Erkenntnisse über die Art und Weise, wie die Softwareschnittstelle die Kommunikation mit einer Applikationssoftware herstellt, genutzt. Aufbauend auf den Ergebnissen des Revers Engineering, das quasi ein Negativ der Softwareschnittstelle abbildet und verwirklicht, wird damit wird einen Teil der geschützten Erfindung bzw. des Verfahrens von neuem hergestellt. Diese Konstellation kann mit dem Umbau eines geschützten Erzeugnisses gleichgestellt werden. Der Dritter benutzt daher eine unter Patentschutz stehende computerimplementierte Erfindung, wenn er auch nur einen Teil des Erzeugnisses oder Verfahrens benutzt, da die Softwareschnittstelle unselbständiger Teil des Schutzgegenstandes ist.
- Im Regelfall wird das Reverse Engineering zu gewerblichen Zwecken durchgeführt, weshalb Interoperabilität nur über § 11 PatG hergestellt werden kann, wenn die durchgeführten Versuche dem Forschungsprivileg zugeordnet werden können.

9.4.2.3 Exkurs: Grundsätze des Reverse Engineering im englischen Recht

Die nationalen Patent- und Urheberrechte der Mitgliedsstaaten der EU kommen zu keinem hiervon wesentlich abweichenden Ergebnissen. Da es sich um einen auf europäischer Ebene

²¹⁹ *Schramm* (Hrsg.), *Der Patentverletzungsprozeß*, 4. Aufl., S. 113 m.w.N.

²²⁰ *Weyand/Haase*, GRUR 2004, 198 (201).

²²¹ *Weyand/Haase*, a.a.O.

²²² *Weyand/Haase*, a.a.O.

harmonierten Bereich handelt, sei stellvertretend an dieser Stelle die Rechtslage nach englischem Recht verwiesen.

Ebenso wenig wie im deutschen Patentrecht, so spricht auch der einschlägige Patents Act 1977 die Frage der Herstellung von Interoperabilität mittels Reverse Engineering nicht unmittelbar und ausdrücklich an. Ob die Durchführung eines Reverse Engineering eine zulässige Benutzungshandlung ist, bestimmt sich nach Sec. 60 Patents Act 1977. Auch im Anwendungsbereich des Patents Act 1977 ist fraglich, inwieweit ein Reverse Engineering patentrechtlich der Neuherstellung des „product“ (Erzeugnis) gleichkommt. Die einschlägige Regelung für ein Erzeugnispatent in Sec. 60 (1)(a) Patents Act 1977 lautet wie folgt:

„ Subject to the provisions of this section, a person infringes a patent for an invention if, but only if, while the patent is in force, he does any of the following things in the United Kingdom in relation to the invention without the consent of the proprietor of the patent, that is to say

(a) where the invention is a product, he makes, disposes of, offers to dispose of, uses or imports the product or keeps it whether for disposal of otherwise;

Die einer Neuherstellung („*make the product*“) gleichkommende Benutzung einer Erfindung löst mithin eine Patentverletzung nach Sec. 60 (1)(a) Patents Act 1977 aus. Durch die Rechtsprechung wurde dieser Regelungsansatz für Fälle der Reparatur eines Erzeugnisses begrenzt, indem dem Erwerber eine sog. „*implied license to repair*“, d.h. stillschweigend erteilte Lizenz zugestanden wurde, die inhaltlich auf die Durchführung von Maßnahmen der Reparatur begrenzt war²²³. Das House of Lords kam in Sachen *United Wire Ltd v Screen Repair Services (Scotland) Ltd*²²⁴ zu einer anderen Beurteilung. Ein solcher Ausnahmetatbestand sei dem Patentrecht fremd, da in jedem Fall zu entscheiden sei, ob es durch die vorgenommenen Handlungen zu einer Neuherstellung des Erzeugnisses komme. Vielmehr handele es sich in diesen Fällen um ein „Nebenrecht“ das im Zusammenhang mit der Berechtigung steht, über das Produkt frei zu verfügen. Vielmehr kommt es auf die Abgrenzung zwischen Reparatur, Ausbesserung und Neuherstellung im Einzelfall an²²⁵. Im Ergebnis unterscheidet sich die Regelung in Sec. 60 (1)(a) Patents Act 1977 von der in § 9 PatG in ihrer systematischen Vorgehensweise nicht. In beiden Rechtsordnungen ist die Frage zu stellen, ob die vom Benutzer vorgenommene Handlung als Reparatur, d.h. als Substanz erhaltend, oder weitergehend der Neuherstellung gleichkommt. Entscheidend ist, dass im Rahmen der Handlungen die durch das Patent geschützten Lehren zum technischen Handeln eingesetzt werden („*a product is made which contains all the essential features claimed*“) ²²⁶. Schon die Verlängerung des Lebenszyklus eines Produktes durch eine Reparatur oder Ausbesserung kann im Einzelfall als Neuherstellung bewertet werden („*to make a product anew*“) ²²⁷.

Beiden Rechtsordnungen ist das hohe Maß an Rechtsunsicherheit in dieser Frage der Abgrenzung von zulässiger und unzulässiger Benutzung in Fallgestaltungen der Wartung und Pflege gemein, da die fallbezogene Abgrenzung der angesprochenen Benutzungshandlungen durch die Rechtsprechung noch nicht zu belastbaren Leitlinien geführt hat.

Wirkungen des Patents werden durch Sec. 60 (5) Patents Act 1977 beschränkt, die ähnlich wie die Regelung in § 11 PatG gestaltet ist. Keine Patentverletzung resultiert danach u.a. aus Handlungen die im privaten Bereich zu nicht gewerblichen Zwecken („*private use for purposes which are not commercial*“) und zu Versuchszwecken („*experimental use relating to the subject-matter*“)

²²³ Solar Thomson Engineering Co. Ltd. v. Barton [1977] RPC 537.

²²⁴ United Wire Ltd. v. Screen Repair Services (Scotland) Ltd. [2000] 4 All ER 353.

²²⁵ United Wire Ltd. v. Screen Repair Services (Scotland) Ltd. [2000] 4 All ER 353 (358 (f, g)).

²²⁶ *Cornish/ Llewelyn*, Intellectual Property, 6 – 09.

²²⁷ *Cornish/ Llewelyn*, Intellectual Property, 6 – 11 mit zahlreichen Rechtsprechungsnachweisen, u.a. auch auf US-amerikanische Rechtsprechung.

of the invention“) vorgenommen werden, von den Patentwirkungen ausgenommen werden. Wie auch in Deutschland wird der Reichweite des Forschungsprivileges begrenzt. Basisforschung, die der Wissenserweiterung dient, wird im Regelfall nicht als patentverletzend angesehen werden. Sobald aber wirtschaftliche Zwecke, zu denen auch die Beschleunigung und Verbesserung eigener Praxisanwendungen („*practical application*“) zu zählen sein wird²²⁸.

Die gleichen Grundsätze finden auch auf Verfahrenspatente nach Sec. 60 (1)(b) Patents Act 1977 (“*a process is infringed directly when the complete process is carried out*“) Anwendung²²⁹.

Urheberrechtlich ist die Dekompilierung von Software in den Grenzen der Regelungen der EG-Richtlinie über den Rechtsschutz von Computerprogrammen vom 14. Mai 1991²³⁰ möglich, die durch den Copyright, Designs and Patents Act 1988 umgesetzt wurde.

9.4.3 Reverse Engineering nach U.S.-amerikanischem Recht

Der Patentinhaber erhält durch 35 U.S.C.A. § 154 das ausschließliche Recht zugewiesen, die geschützte Erfindung herzustellen, zu benutzen und zu verkaufen. Hierin ist zugleich auch ein negatives Ausschlussrecht gegenüber Dritten enthalten.

Auch das US-amerikanische Patentrecht unterscheidet hinsichtlich der Zulässigkeit von Benutzungshandlungen zwischen zulässigem reparieren und nicht zulässiger Rekonstruktion. Der an sich in den U.S.A. geltende Erschöpfungsgrundsatz wird insoweit eingeschränkt²³¹. Immer dann, wenn die Benutzung der patentrechtlich geschützten Lehren im Sinne eines „*new making of the item*“ kommt, sind derartige Handlungen nicht von dem Benutzungsrecht des Erwerbers eines Produkts erfasst²³². Auch nach U.S.-amerikanischem Patentrecht bestehen nicht unerhebliche Abgrenzungsprobleme. Überwiegt der wirtschaftliche Wert des ersetzten Teils des Erzeugnisses, wird in solchen Fällen im Zweifel eine Neuherstellung angenommen werden. Ähnlich werden die Fälle der Verlängerung der Lebenszyklen von Erzeugnissen behandelt²³³.

Beschränkungen der Wirkung des Patents, wie sie in § 11 PatG oder Sec. 60 (5) Patents Act 1977 vorgesehen sind, kennt das US-amerikanische Patentrecht nicht. Durchbrechungen des Ausschließlichkeitsrechts kann lediglich über Missbrauchstatbestände erreicht werden, und zwar wenn der Patentinhaber sein Monopolrecht über seine gesetzlich zugewiesenen Befugnisse hinaus ausdehnt (sog. „*misuse doctrine*“). Die notwendigen Korrekturen sind jedoch außerhalb des Patentrechts vorzunehmen und erfolgen durch die Anwendung kartellrechtlicher Grundsätze (*antitrust law*). Hierbei können U.S.-amerikanische Entscheidungen herangezogen werden, die den in diesem Gutachten unter dem Blickwinkel der Herstellung von Interoperabilität ähnlich sind. Wird ein Patent dazu benutzt, Märkte mittels des Patents zu monopolisieren, die nicht vom Patent erfasste Komponenten betreffen, wird Missbrauch i.S.d. „*misuse doctrine*“ bejaht²³⁴. Im Ergebnis hat dieser Regelungsansatz, insbesondere vor dem Hintergrund der in den U.S.A. sehr weitgehenden Patentierung von Software, zu erheblichen Verwerfungen geführt, deren gesetzgeberische Korrektur in letzter Zeit vermehrt gefordert wird. Insbesondere die Tendenz der Vernetzung von Patenten durch „Cross-Lizenz-Vereinbarungen“ und die Patent-

²²⁸ Cornish/ Llewelyn, Intellectual Property, 6 – 11 m.w.Nw.

²²⁹ Cornish/ Llewelyn, Intellectual Property, 6 – 09, 6 - 10.

²³⁰ ABL. EG Nr. 1122/42.

²³¹ Miller/ Davis, Intellectual Property, p. 135.

²³² *Cinema Patents Co. v. Craft Film Laboratories, Inc.*, 56 F.2d 265 (D.C.Del. 1932), affirmed 64 F.2d 42.

²³³ *Aro Manufacturing Co. v. Convertible Top Replacement Co.*, 365 U.S. 336.

²³⁴ *Rohm and Haas Co. v. Dawson Chemical Co.*, 599 F.2d 685 (5th. Cir. 1979), affirmed 100 S.Ct. 2601.

pool-Bildung setzt marktführende Unternehmen in die Lage, vermehrt nicht nur die Märkte, auf denen sie wirtschaftlich tätig sind, zu monopolisieren²³⁵.

Die Herstellung von Interoperabilität wird in den U.S.A. ebenfalls über urheberrechtliche Grundsätze gewährleistet. Es ist nur für „legitime Zwecke“ zulässig²³⁶. Ob Dekompilierung auch in patentrechtlichen Fallgestaltungen möglich ist, wird in den U.S.A. diskutiert, gerichtliche Entscheidungen liegen jedoch noch nicht vor²³⁷. Eine der Strategien, die U.S.-amerikanische Softwareunternehmen gegen die Möglichkeit des Reverse Engineering einzusetzen versuchen, ist die Vereinbarung von Klauseln in Lizenzverträgen, die das Verbot der Durchführung eines Reverse Engineering vorsehen. Die Durchsetzbarkeit solcher Klauseln ist auch in den U.S.A. noch umstritten. War vom California Supreme Court der Fall *DVD CCA v. Bunner*²³⁸ zu entscheiden, der in den U.S.A. große Aufmerksamkeit auf sich gezogen hatte, da dieser die Frage der Zulässigkeit derartiger Lizenzklauseln betraf. Der Beklagte hatte u.a. im Internet Informationen für die Dekodierung von DVD's veröffentlicht. Der Rechtsstreit wurde zur weiteren Verhandlung an den California Court of Appeal zurückverwiesen. Wie sich die weitere Rechtsentwicklung in dem Mehrrechtsstaat U.S.A. entwickeln wird zurzeit ist noch nicht absehbar. Sicher ist nur, kurzfristig keine Beruhigung der Diskussion durch höchstgerichtliche Entscheidungen erwartet werden kann. Im Ergebnis ist danach mit Blick auf die Zulässigkeit der Durchführung von Reverse Engineering – Handlungen nach wie vor kein ausreichendes Maß an Rechtssicherheit gegeben.

9.4.4 Reverse Engineering im indischen Immaterialgüterrecht

Eine explosionsartige Entwicklung hat der Softwareentwicklungsmarkt in Indien genommen. Nachdem eine Reihe inländischer Unternehmen wegen Patentverletzungen von ausländischen Patentinhabern verklagt wurden, hat Indien, aus dem Gesichtspunkt des Inländerschutzes das Patentrecht novelliert und dabei den Patentschutz gelockert²³⁹. Softwareschutz wird jedoch primär über das Urheberrecht erreicht den „Indian Copyright Act 1957. Sec. 52 des Indian Copyright Act 1957 enthält eine Auflistung von Nutzungshandlungen, die nicht zu einer Verletzung des Urheberrechts führen. Dort werden in Sec. 52 (1)(a) zwei Fallvarianten angesprochen und zwar die private Nutzung sowie Forschung und, übertragen auf Software, die Dekompilierung. Diese verbirgt sich hinter dem Begriff des „review“, da die nachfolgenden Spezifizierungen unter anderem auf die Anpassung von Software Bezug nehmen.

Das Patentrecht ist gleichfalls gesetzlich geregelt, und zwar durch den Patents Act 1970. Schon seit Jahren gibt der Patents Act 1970 Anlass zu kontroversen Diskussionen, da er bestimmte Erzeugnisse vom Patentschutz ausnimmt. Diese Ausnahmetatbestände ergeben sich aus Sec. 3 Patents Act 1970 zu denen auch pharmazeutische Produkte zählen. In Sec. 48 werden die Wirkungen eines erteilten Patents beschrieben, wobei berücksichtigt werden muss, dass ein Patent unter bestimmten, in Sec. 47 aufgeführten Auflagen erteilt werden kann, die in groben Zügen an die Regelungssystematik des § 11 PatG erinnern. Eine Beschränkung der Patentwirkungen werden u.a. die „eigene Nutzung“ (Sec. 47 (1)) und „Forschung und Lehre“ (Sec. 47 (3)) vorgesehen. Spezielle Regelungen, die die Herstellung von Interoperabilität bezwecken, sind

²³⁵ Vgl. *Shapiro*, Navigating the Patent Thicket: Cross Licenses, Patent Pools, and Standard-Setting, 2001 –<http://haas.berkeley.edu/~shapiro/thicket/pdf>. (letzter Abruf 3. Oktober 2004).

²³⁶ *Sega Enterprises Ltd. v. Accolade, Inc.*, 977 F.2d 1510 (9th. Cir. 1992), ausführlich hierzu *Passavant* in: Ullrich/Körner, Der internationale Softwarevertrag, II Rdnr. 1824ff.

²³⁷ *Cohen/ Lemley*, 89 Calif.L.Rev. 1, 6 (2000); *Samuelson/ Scotchmer*, The Law & Economics of Reverse Engineering, 111 Yale.L.J. 1575.

²³⁸ *DVD CCA v. Bunner*, Calif.Sup.Ct. (Urteil vom 25. August 2003, letzter Abruf 5. Oktober 2004) <http://news.findlaw.com/hdocs/docs/dvd/dvdccabnr82503opn.pdf>.

²³⁹ *Williamson*, The International Enforcement of Software Copyrights and Patents, S. 74.

auch im Indian Patent Act 1970 nicht enthalten. Im Gegensatz etwa zum deutschen PatG wird die Erteilung von Zwangslizenzen („compulsory license“, Sec. 82 – 98) in für europäische Verhältnisse ungewöhnlicher Tiefe geregelt. An dieser Stelle kommt die Interessenausrichtung eines Schwellenlandes zum Ausdruck, zumal gerade Indien Bedürfnisse der Entwicklungsländer im Rahmen der TRIPS-Verhandlungen mit besonderem Nachdruck vertreten hat. Ausdrückliche Regelungen, die die Fallgestaltungen des Reverse Engineering ansprechen, enthält der Indian Patent Act 1970 nicht. Im Ergebnis kann somit festgestellt werden, dass dem indischen Immaterialgüterrecht nicht ausdrücklich entnommen werden kann²⁴⁰. Es verbliebe damit allein die Möglichkeit entsprechende Nutzungs- oder Benutzungsrechte im Wege vertraglicher Vereinbarung einzuräumen²⁴¹.

9.4.5 Rechtsvergleichendes Fazit zum Reverse Engineering

Regelungen, die Fallgestaltungen der Herstellung von Interoperabilität, wie etwa die Dekompilierung, finden sich jeweils nur in der urheberrechtlichen Gesetzgebung. Die exemplarisch untersuchten Patentgesetze der hier referierten Rechtsordnungen enthalten, mit Ausnahme des U.S.-amerikanischen Patentrechts Regelungen, die für die Fälle der privaten Benutzung und zu Zwecken der Forschung, die Wirkungen eines erteilten Patents begrenzen. In den U.S.A. wird die monopolisierende Wirkung von Patenten lediglich in Missbrauchsfällen durch die Anwendung kartellrechtlicher Grundsätze begrenzt.

Da Softwareschutz bis dato in erster Linie über die Anwendung des Urheberrechts erfolgt ist, bestand nur in diesem Bereich ein Regelungsbedarf. Anders lässt sich die Divergenz zwischen Urheber- und Patentrecht nicht erklären. Soweit in dem für die Erstellung dieses Gutachtens gesetzten Zeitrahmen die Behandlung von Maßnahmen des Reverse Engineering ermittelt werden konnte, hat sich ergeben, dass nur solche Handlungen zulässig sind, die faktisch nicht zu einer Neuherstellung des Erzeugnisses oder Verfahrens führen. Die in diesen Fallgestaltungen auftretenden Interessenkollisionen liegen gleich wohl auf der Hand. Gerade im Bereich der Softwareerstellung eröffnet das Reverse Engineering oder die Dekompilierung Wettbewerbern von Softwareherstellern relativ einfachen Zugang zu Informationen, die eine Imitierung der analysierten Software zulassen. Nicht zuletzt deshalb hat z.B. das australische „Copyright Law Review Committee“ schon im Jahre 1995 die Empfehlung ausgesprochen, im australischen Urheberrecht Maßnahmen des Reverse Engineering und der Dekompilierung als grundsätzlich verbotene Nutzung zu qualifizieren und schlägt vor, Ausnahmen nur für begrenzte Fallgestaltungen vorzusehen, zu denen u.a. die Fälle der Herstellung von Interoperabilität sowie des Debugging zählen²⁴².

Diese Tendenz wird offenbar durch das TRIPS-Übereinkommen in besonderer Weise gefördert. Als Beispiel mag an dieser Stelle das russische Immaterialgüterrecht dienen, das an die durch das TRIPS-Übereinkommen vorgegebenen Mindeststandards angepasst wurde. Russlands Patentgesetz ist am 14. Oktober 1992 in Kraft getreten. Aus Art. 10 des russischen Patentgesetzes folgen die Wirkungen des Patents die sich in etwa an den Vorgaben des TRIPS-Übereinkommen ausrichten. Handlungen die keine Patentverletzung darstellen, werden in Art. 11 aufgelistet. In der Auflistung finden sich auch die private Benutzung und solche, die zu nicht-kommerziellen Zwecken erfolgt, womit das Forschungsprivileg angesprochen wird. Wie auch in den anderen untersuchten Rechtsordnungen, enthält das russische Patentgesetz keine ausdrücklichen Regelungen über die Zulässigkeit eines Reverse Engineering.

²⁴⁰ Ebenso *Dohrmann*, S.144f.

²⁴¹ *Dohrmann*, S.147.

²⁴² <http://www.itee.uq.edu.au/cristina/cj96.ps>. (letzter Abruf 5. Oktober 2004).

9.5 Herstellung von Interoperabilität durch Zwangslizenzen nach § 24 PatG

Durch Zwangslizenz kann im Einzelfall unter den Voraussetzungen des § 24 PatG die nicht ausschließliche Befugnis zur gewerblichen Nutzung einer Erfindung erteilt werden²⁴³. Die Regelung entspricht den Anforderungen des Art. 31 TRIPS-Übereinkommen²⁴⁴. Nach aktueller Rechtslage müssen folgende Voraussetzungen erfüllt sein, damit eine Zwangslizenz erteilt werden kann:

- der Lizenzsucher hat seine Bereitschaft erkennen lassen müssen, eine geschäftsübliche Vergütung zu zahlen,
- für die Vergütung muss eine Sicherheitsleistung angeboten werden,
- der Patentinhaber muss eine gütliche Zustimmung zur Benutzung des Patents verweigert haben und
- es muss, wegen des Eingriffs in Art. 14 GG, ein öffentliches Interesse vorliegen, wobei der Grundsatz der Verhältnismäßigkeit zu berücksichtigen ist.

Nur in besonderen Ausnahmefällen lässt das Patentrecht die Aufbrechung des Ausschließungsrechts des Patentinhabers zu, und zwar dann, wenn ein Missbrauchsfall vorliegt, d.h. er durch die Verweigerung der Lizenzerteilung den technischen Fortschritt oder die wirtschaftliche Entwicklung in untragbarer Weise hemmt bzw. der Befriedigung wesentlicher sozialer Bedürfnisse entgegensteht²⁴⁵. Beispiele hierfür sind: Versorgung der Bevölkerung mit wichtigen Bedarfsgütern, Beschaffung dringend benötigter Rohstoffe, Erhöhung der Betriebssicherheit und Förderung der Gesundheitspflege.

Praktische Bedeutung haben Zwangslizenzen nicht in nennenswertem Umfang erlangt. Lediglich in einem Fall, es ging um die Erteilung einer Zwangslizenz für ein Patent auf menschliches Interferon, wurde in Deutschland eine solche erteilt. Der BGH hob ein die Erteilung bestätigendes Urteil jedoch wieder auf. Ein deutsches Pharmaunternehmen war Inhaberin eines Patents zur Herstellung des Wirkstoffs Interferon-gamma, beabsichtigte aber nicht die von ihr hergestellten Medikamente weiterzuentwickeln. Ein Konkurrenzunternehmen hatte ein Interferon-Medikament für eine neue Indikation (Rheuma) entwickelt, für das bereits die Marktzulassung erteilt worden war. Da die Patentinhaberin keine Lizenzvereinbarung abschließen wollte, wurde schließlich eine Zwangslizenz erteilt, die das BPatG²⁴⁶ noch bestätigte, aber vom BGH aufgehoben wurde²⁴⁷. Über diesen Anwendungsfall hinaus wurden zwar weitere Anträge vereinzelt gestellt, es kam aber in keinem dieser Fälle zur Erteilung einer Zwangslizenz²⁴⁸.

²⁴³ § 24 Abs. 1 PatG, Die nicht ausschließliche Befugnis zur gewerblichen Benutzung einer Erfindung wird durch das Patentgericht im Einzelfall nach Maßgabe der nachfolgenden Vorschriften erteilt (Zwangslizenz),
1. der Lizenzsucher sich innerhalb eines angemessenen Zeitraumes erfolglos bemüht hat, vom Patentinhaber die Zustimmung zu erhalten, die Erfindung zu angemessenen geschäftsüblichen Bedingungen zu benutzen und
2. das öffentliche Interesse die Erteilung einer Zwangslizenz gebietet.

²⁴⁴ Im Rahmen der Diskussion über die Beseitigung von Handelshemmnissen unter WTO-Recht, spielt die Frage der Erteilung von Zwangslizenzen häufig keine, oder nur eine untergeordnete Rolle, vgl. *Fischer*, 100f. der dieses Thema im Rahmen der Darstellung von Informationsverfahren über technische Normen und Harmonisierungsansätze ebenfalls nicht anspricht.

²⁴⁵ *Kraßer*, Patentrecht, S. 860.

²⁴⁶ *BPatGE* 32, 184.

²⁴⁷ *BGHZ* 131, 247.

²⁴⁸ Vgl. *Kraßer*, Patentrecht, S. 861 m.w.Nw.

In zwei speziellen Bereichen des Patentrechts wird die Frage der Erteilung von Zwangslizenzen gesondert angesprochen. Es handelt sich um Fälle der Halbleitertechnologie sowie der Biotechnologie und Gentechnik. Für das Gebiet der Halbleitertechnologie werden die Voraussetzungen der Erteilung verschärft, in dem sie nur in Fällen erteilt werden darf, wenn dies zur Behebung einer in einem Gerichts- oder Verwaltungsverfahren festgestellten wettbewerbswidrigen Praxis des Patentinhabers erforderlich ist (§ 24 Abs. 3 PatG). Voraussetzung für die Erteilung einer Zwangslizenz in diesem Bereich ist somit der rechtskräftige Abschluss eines kartellrechtlichen Verfahrens. Im Zuge der Umsetzung der Richtlinie über den Schutz biotechnologischer Erfindungen (BioPat-RL)²⁴⁹, plant die Bundesregierung die Anpassung des PatG und beabsichtigt in § 24 Abs. 2 PatG das Erfordernis des öffentlichen Interesses an der Erteilung einer Zwangslizenz für abhängige Patente, entsprechend Art. 12 BioPat-RL, fallen zu lassen. In den Erwägungsgründen der Richtlinie findet sich hierzu folgende Begründung:

„(52) Für den Bereich der Nutzung der auf gentechnischem Wege erzielten neuen Merkmale von Pflanzensorten muss in Form einer Zwangslizenz gegen eine Vergütung ein garantierter Zugang vorgesehen werden, wenn die Pflanzensorte in Bezug auf die betreffende Gattung oder Art einen bedeutenden technischen Fortschritt von erheblichem wirtschaftlichem Interesse gegenüber der patentgeschützten Erfindung darstellt.

(53) Für den Bereich der gentechnischen Nutzung neuer, aus neuen Pflanzensorten hervorgegangener pflanzlicher Merkmale muss in Form einer Zwangslizenz gegen eine Vergütung ein garantierter Zugang vorgesehen werden, wenn die Erfindung einen bedeutenden technischen Fortschritt von erheblichem wirtschaftlichem Interesse darstellt.“

Eine richtlinienkonforme Umsetzung in nationales Recht hätte, würde man Zwangslizenzen als Abhängigkeitslizenzen für biotechnologische Erfindungen unter geringeren Voraussetzungen als auf anderen Gebieten der Technik erlangen können, hätte die Beibehaltung der Voraussetzung des Bestehens eines öffentlichen Interesses zu nicht gerechtfertigten unstimmgigen Ergebnissen geführt²⁵⁰. Nachzuweisen bleibt von dem eine Zwangslizenz Nachsuchenden, dass seine Erfindung einen bedeutenden technischen Fortschritt von erheblichem wirtschaftlichen Interesse darstellt. Erläuterungen, welche Voraussetzungen zur Erfüllung dieses unbestimmt formulierten Merkmals vorliegen müssen, enthalten weder die BioPat-RL noch der Gesetzesentwurf zur Änderung des PatG und eröffnet weite Auslegungsspielräume. Da eine Interessenabwägung zwischen den Belangen des Patentinhabers und der Lizenzsuchenden zu erfolgen hat, besteht nach wie vor eine erhebliche Hürde, da es sich um einen grundrechtsrelevanten Eingriff in das Eigentumsrecht des Patentinhabers handelt. Mit Ausnahme der Entscheidung des BGH „Interferon-gamma“, hat sich die Rechtsprechung in Deutschland nicht mit der Frage der Erteilung von Zwangslizenzen zu befassen. Die Eignung des Mittels „Zwangslizenz“ zur Herstellung von Interoperabilität kann folglich zurzeit nicht belastbar beantwortet werden²⁵¹.

9.6 Herstellung von Interoperabilität durch das Wettbewerbsrecht

Wettbewerbsrechtliche Implikationen folgen bereits aus dem Ausschließlichkeitsrecht, welches das Patentrecht gewährt. Ferner sind sog. Sperrpatente, d.h. solche, die ausschließlich der Unterbindung fremder Anwendungstätigkeit dienen sowie Patenthäufungen, also die Erlangung einer Vielzahl von Patenten auf einem Gebiet der Technik geeignet, zu einer **Marktbeherr-**

²⁴⁹ Richtlinie 98/44/EG des Europäischen Parlamentes und des Rates vom 6. Juli 1998 über den rechtlichen Schutz biotechnologischer Erfindungen, ABl. EG Nr. L 213 S. 13.

²⁵⁰ Entwurf eines Gesetzes zur Umsetzung der Richtlinie über den rechtlichen Schutz biotechnologischer Erfindungen, BT-Dr. 15/1709, S. 13.

²⁵¹ Nach wie vor Novellierungsbedarf feststellend, *Scheffler*, GRUR 2003, 97 (105), der derzeit die Erhebung einer Patentnichtigkeitsklage als sehr viel wirkungsvoller erachtet (101).

schung zu führen und damit wettbewerbsrechtliche Folgen auszulösen²⁵². Auch Patentgemeinschaften, sog. „patent pools“, und „Cross-License-Vereinbarungen“ vermögen kartellähnliche Wirkungen hervorzurufen, da sie zu Konzentrationen führen, die im Einzelfall unter gesamtwirtschaftlicher Betrachtungsweise nicht erwünscht sind²⁵³. Zu berücksichtigen ist weiter die Tatsache der Abhängigkeit von Unternehmensgröße und Patentbesitz. Da größere Unternehmen sehr viel mehr in der Lage sind Forschungs- und Entwicklungstätigkeit zu finanzieren, ergibt sich hieraus zwangsläufig eine Scherenentwicklung²⁵⁴. Wenn Microsoft eine Patentoffensive ankündigt, spricht diese Tatsache für sich und lässt vermuten, dass man eine größtmögliche Abdeckung des Technologiebereiches „Software“ anstrebt²⁵⁵. Auch infolge der Ankündigung der Einigung in einem Patenstreit zwischen Sun und Microsoft, sind Befürchtungen entstanden, es könne nunmehr wegen der zu erwartenden Kooperation und gegenseitigen Lizenzerteilungen mit wettbewerbsbeeinträchtigenden Verhaltensweisen gerechnet werden²⁵⁶. Ein weiterer Gesichtspunkt ist in der vom niederländischen Wirtschaftsministeriums beauftragten und von PwC erstellten Studie unter dem Titel „Rethinking the European ICT Agenda – Ten ICT-breakthroughs for reaching Lisbon goals“ herausgearbeitet worden. Patente werden nach der Studie als Markteintrittsbarrieren betrachtet, die insbesondere für Unternehmensneugründungen ein wirtschaftliches und damit erfolgreiches Agieren auf dem Informations- und Kommunikationsmarkt erschweren²⁵⁷.

Störungen des Marktes kann nur über das Kartellrecht begegnet werden, d.h. das Wettbewerbsrecht im weiteren Sinne. Das Lauterkeitsrecht des UWG scheidet aus, da es nicht dem Abbau von Handelshemmnissen und der Marktöffnung dient, sondern Regelungen enthält, die auf ein allgemeines Unternehmensverhaltensrecht im Wettbewerb im Rahmen der unternehmerischen Tätigkeit gerichtet sind, also Handlungen im geschäftlichen Verkehr²⁵⁸.

In den §§ 17 und 18 GWB sind Regelungen über Wettbewerbsbeschränkungen in Verträgen über Veräußerung, Lizenzierung oder Überlassung zur Nutzung von gewerblichen und sonstigen Schutzrechten, technischem Know-how sowie Saatgut enthalten.

Eine unzulässige Beschränkung liegt vor, wenn man aufgrund einer Gesamtbetrachtung aller vertraglichen Rechte und Pflichten zu dem Ergebnis gelangt, dass dem Lizenznehmer Wettbewerbsverbote auferlegt werden. Wird kein Patent erteilt, sind die Regelungen nicht anwendbar, da sie sich lediglich gegen die vertragliche Ausdehnung von Schutzrechtswirkungen über den gesetzlichen Rahmen hinaus wenden. In derartigen Fallgestaltungen besteht nur die Möglichkeit auf die Grundsätze über den Behinderungswettbewerb, d.h. der Missbrauchsaufsicht gegenüber marktbeherrschenden Unternehmen²⁵⁹, zurückzugreifen. Hierunter kann auch die Lieferverweigerung, d.h. die Nichterteilung einer Lizenz, gefasst werden. Voraussetzung ist in die-

²⁵² Kraßer, Patentrecht, S. 46.

²⁵³ Kraßer, Patentrecht, S. 47.

²⁵⁴ Ausführlich und instruktiv zum dem Spannungsfeld von Kartellrecht und Immaterialgüterschutz von Erfindungen aus dem IT-Bereich, Wolf, S. 39ff.

²⁵⁵ <http://www.microsoft.com/presspass/press/2004/jul04/07-29FAM2004PR.asp> (letzter Abruf 23. September 2004).

²⁵⁶ <http://www.heise.de/newsticker/meldung/51137> (letzter Abruf 23. September 2004).

²⁵⁷ Rethinking the European ICT Agenda – Ten ICT-breakthroughs for reaching Lisbon goals, S. 50 Ziff. 342, <http://www.ez.nl/content.jsp?objectid=24583> (letzter Abruf 17. September 2004).

²⁵⁸ Emmerich, Unlauterer Wettbewerb, S. 20f.; Gassner, Grundzüge des Kartellrechts, S. 1.

²⁵⁹ Der Begriff der Marktbeherrschung wird nicht einheitlich verstanden (sog. „Relativität des Marktbeherrschungsbegriffs“) und wird teilweise, je nach Zweck der anzuwendenden Eingriffsnorm, unterschiedlich interpretiert; Gassner, Grundzüge des Kartellrechts, S. 102f. m.w.Nw. In diesem Gutachten wird von einem neoklassischen Begriffsverständnis ausgegangen, d.h. dass als Hauptaufgabe der Wettbewerbspolitik die Offenhaltung der Märkte verstanden wird; vgl. Emmerich, Kartellrecht, S. 8f.; Gassner, Grundzüge des Kartellrechts, S. 5.

sen Fällen die Feststellung der Unbilligkeit des Verhaltens nach § 20 GWB. Unbilligkeit ist dann anzunehmen, wenn kein sachlicher Grund vorhanden ist, der das Verhalten des Schutzrechtsinhabers zu rechtfertigen mag.

9.6.1 Patentlizenzverträge

Patentlizenzverträge fallen in den Anwendungsbereich des § 17 GWB. Systematisch sind die §§ 17 und 18 GWB den Regelungen über Vertikalvereinbarungen zuzuordnen, betreffen somit nur Vereinbarungen von Unternehmen, die miteinander auf dem gleichen Markt im Wettbewerb stehen.

Zentrale Bedeutung kommt der Regelung des Abs. 1 zu:

„Verträge über Veräußerung oder Lizenzierung von erteilten oder angemeldeten Patenten oder Gebrauchsmustern, von Topographien oder Sortenschutzrechten sind verboten, soweit sie dem Erwerber oder Lizenznehmer Beschränkungen im Geschäftsverkehr auferlegen, die über den Inhalt des gewerblichen Schutzrechts hinausgehen. Beschränkungen hinsichtlich Art, Umfang, technischem Anwendungsbereich, Menge, Gebiet oder Zeit der Ausübung des Schutzrechts gehen nicht über den Inhalt des Schutzrechts hinaus.“

Unerheblich ist, ob es sich um ein erteiltes oder lediglich angemeldetes Patent handelt. Auch die Wirksamkeit des Schutzrechts ist ohne Bedeutung²⁶⁰. Verträge über wertlose Scheinpatente können keine Wettbewerbsbeschränkungen zu Lasten des Lizenznehmers rechtfertigen, da in diesen Fällen kein Raum für die Anwendung des § 17 GWB verbleibt²⁶¹. Nur im Fall der spürbaren Beschränkung des Lizenznehmers werden die Rechtsfolgen des § 17 Abs. 1 GWB ausgelöst²⁶². Sie müssen daher über Berichtserweiterte Schutzpflichten, wie der Mitteilung der dem Lizenznehmer bekannt gewordenen Schutzrechtsverletzungen Dritter, deutlich hinausgehen.

Anwendungsfälle aus dem Bereich der Informationstechnologie können sog. „Trivialpatente“ sein. Zur Klärung der Anwendbarkeit des § 17 GWB wird es nach den Erfahrungen aus der Praxis jedoch aller Voraussicht nach nicht kommen, da der marktmachtschwächere der Beteiligten im Regelfall der Lizenznehmer ist und dieser sich nicht in der Lage sieht, gegen verbotene Verhaltensweisen nach § 17 GWB vorzugehen.

Keine unzulässige Beschränkung wird nach § 17 Abs. 1 Satz 2 GWB in Beschränkungen des Lizenznehmers hinsichtlich Art, Umfang und technischem Anwendungsgebiet der Ausübung des Schutzrechts gesehen. Dem Patentinhaber ist damit die Möglichkeit eröffnet, einzelne in § 9 PatG genannte Benutzungsarten auszuschließen, wie z.B. die Eingrenzung auf Herstellungs-, Gebrauchs- oder Vertriebslizenzen²⁶³.

§ 17 Abs. 2 GWB sieht einen Katalog von Lizenzbeschränkungen vor, die selbst dann zulässig sind, wenn sie im Einzelfall über den Inhalt des betreffenden Schutzrechtes hinausgehen und an sich gegen § 17 Abs. 1 GWB verstoßen würden. Absatz 1 gilt danach nicht für den Erwerber oder Lizenznehmer beschränkende Bindungen,

„1. soweit und solange sie durch ein Interesse des Veräußerers oder Lizenzgebers an einer technisch einwandfreien Ausnutzung des Gegenstandes des Schutzrechts gerechtfertigt sind (technisch bedingte Beschränkungen),

2. die zum Erfahrungsaustausch oder zur Gewährung von nicht ausschließlichen Lizenzen auf Verbesserungs- oder Anwendungserfindungen verpflichten, sofern diesen gleichartige Verpflichtungen des Veräußerers oder Lizenzgebers entsprechen (Rückgewährklauseln),

²⁶⁰ BGHZ 46, 365 (371f.); BGH NJW 1965, 499.

²⁶¹ Emmerich, Kartellrecht, S. 141.

²⁶² Emmerich, Kartellrecht, S. 143.

²⁶³ BGHZ 52, 55 (57f.) – Frischhaltegefäß; OLG Düsseldorf, GRUR 1981, 919 (921).

3. das lizenzierte Schutzrecht nicht anzugreifen (Nichtangriffsklauseln²⁶⁴),
 4. das lizenzierte Schutzrecht in einem Mindestumfang zu nutzen oder eine Mindestgebühr zu zahlen,
 5. die Lizenzzeugnisse in einer den Herstellerhinweis nicht ausschließenden Weise zu kennzeichnen,
- soweit diese Beschränkungen die Laufzeit des erworbenen oder in Lizenz genommenen Schutzrechts nicht überschreiten.“

Zu beachten ist, dass Nichtangriffsklauseln nur im Verhältnis zum lizenzierten Schutzrecht bestehen. Eine Erstreckung darüber hinaus ist nichtig²⁶⁵, ebenso die das Verbot der Gefährdung des Schutzrechts durch eigene Forschung und Entwicklung²⁶⁶.

9.6.2 Patentgemeinschaften und Cross-Lizenzvereinbarungen

Unter Patentgemeinschaften, sog. „Patentpools“ wird kartellrechtlich ein Zusammenschluss von Schutzrechtsinhabern zwecks Vergemeinschaftung des Patentbesitzes verstanden²⁶⁷. Sie sind nicht nach §§ 17, 18 GWB, sondern wegen ihres Kartellcharakters als Horizontalvereinbarung, nach § 1 GWB zu beurteilen. Anwendungsfälle sind die Gründung von Gemeinschaftsunternehmen durch konkurrierende Unternehmen, wobei die Patente in das Gemeinschaftsunternehmen eingebracht werden und von dem sie sodann Lizenzen in Verbindung mit Mengenbeschränkungen und Preisbindungen erhalten sollen²⁶⁸. Werden im Rahmen der Bildung eines Patentpools gegenseitige Wettbewerbsverbote vereinbart, greift ebenfalls § 1 GWB²⁶⁹. Gleiches gilt für sog. Cross-Lizenz-Vereinbarungen, die über das allgemeine Kartellverbot § 1 GWB, Art. 81 EGV angegriffen werden können.

9.6.3 Behinderung durch Sperrpatente

Werden Patente dazu benutzt, fremde unternehmerische Tätigkeit zu verhindern, kommt die Anwendung des § 20 GWB in Betracht, der Regelungen über das sog. „horizontale Behinderungsverbot“ enthält. Anwendbar ist die Vorschrift nur in Fällen, in denen ein Wettbewerber mit überlegener Marktmacht diese dazu ausnutzt, kleine und mittlere Wettbewerber unmittelbar oder mittelbar unbillig zu behindern. Angesprochen sind Fallgestaltungen relativer horizontaler Übermacht von großen Unternehmen gegenüber kleinen und mittleren Konkurrenten²⁷⁰. Ausschlaggebend ist das Machtgefälle zwischen den betreffenden Unternehmen, mit der Folge, dass der Wettbewerb das Verhalten des großen Unternehmens nicht mehr in ausreichender Weise zu kontrollieren vermag. Entscheidend hierfür ist das Verhältnis der Marktanteile²⁷¹. Kann ein Verstoß gegen § 20 GWB festgestellt werden, besteht für die Kartellbehörden gegen das marktstarke Unternehmen nach § 32 GWB vorzugehen, d.h. das diskriminierende Verhalten zu untersagen. Ein Belieferungspflicht kann auf diesen Wege dem marktstarken Unternehmen allerdings nicht auferlegt werden²⁷². Dieser kann nur auf §§ 20 Abs. 1, 33 GWB i.V.m. § 249 Satz

²⁶⁴ Angesprochen sind Einspruch (§ 59 PatG), Beschwerde (§ 73 PatG) und die Nichtigkeitsklage (§§ 22, 81 PatG).

²⁶⁵ BKartA, Tätigkeitsbericht 1981/82, S. 93.

²⁶⁶ BKartA, Tätigkeitsbericht 1985/86, S. 36; 1987/88, S. 35.

²⁶⁷ *Emmerich*, Kartellrecht, S. 142 m.w.Nw.

²⁶⁸ BKartA, Tätigkeitsbericht 1970, S. 92f.; 1993/94, S. 31, 101 – Flaschenkästen aus Kunststoff.

²⁶⁹ BKartA, Tätigkeitsbericht 1977, S. 90; 1978, S. 100.

²⁷⁰ BT-Dr. 8/3690, S. 23.

²⁷¹ *BGHZ* 129, 203 (209f.)

²⁷² *BGHZ* 67, 104 (107ff.).

1 BGB gestützt werden, wenn nur auf diesem Wege Abhilfe möglich ist. Auch die Erteilung einer Zwangslizenz nach § 24 PatG kann in Betracht kommen, um das Sperrpatent aufzubrechen.

9.6.4 Schutzrechtshäufung

Patenthäufung kann im Fall einer marktbeherrschende Stellung des Unternehmens, das die Patente innehat, als missbräuchliche Ausnutzung nach § 19 GWB, Art. 82 EGV qualifiziert werden.

9.6.5 Anwendungsfall Microsoft

Im Fall Microsoft ist die Europäische Kommission aus zwei Gründen vorgegangen, von denen hier nur einer von Interesse ist. Keine Bedeutung für die Herstellung von Interoperabilität hat die unerlaubte Kopplung des „Windows Media Player“ mit dem Microsoft-Betriebssystem Windows. Relevant ist dagegen das Verhalten der Fa. Microsoft, das als bewusste Einschränkung der Dialogfähigkeit des Windows-Betriebssystems²⁷³.

Rechtsgrundlage des Vorgehens der Europäischen Kommission war Art. 82 EGV, der den Missbrauch einer marktbeherrschenden Stellung verbietet²⁷⁴. Fraglich war, unter welchen Voraussetzungen in einem solchen Fall das marktbeherrschende Unternehmen verpflichtet ist, Wettbewerber zu unterstützen, indem sie diesen Zugang z.B. zu gewerblichen Schutzrechten geben müssen. Bereits in den 80er Jahren gab es einen ähnlich gelagerten Fall. IBM war nicht bereit, Wettbewerbern rechtzeitig technische Informationen zur Verfügung zu stellen, welche diese benötigten, damit ihre Produkte mit dem IBM-System/370 zusammenarbeiten konnten. Es ging also um Schnittstellen- bzw. Interface-Informationen, einen klassischen Fall der Herstellung von Interoperabilität. Zur Entscheidung kam das Verfahren nicht, da IBM sich verpflichtete, die Schnittstellen-Informationen auf Anfrage an Wettbewerber herauszugeben²⁷⁵.

Auf europäischer Ebene hatte sich die Rechtsprechung später mit Fällen des Marktmissbrauchs zu befassen, ohne das dabei jedoch Fallgestaltungen auftraten, die die Frage der Herstellung von Interoperabilität betrafen. Im Rahmen der Auseinandersetzung zwischen der Kommission und Microsoft wird der EuGH-Entscheidung im Fall Magill²⁷⁶ besondere Bedeutung beigemessen. Der Entscheidung lag kurz gefasst folgender Sachverhalt zugrunde:

Einer Fernsehzeitschrift war es nicht möglich, eine umfassende wöchentliche Programmzeitschrift herauszugeben, weil sich die in Irland ansässigen Fernsehanstalten weigerten, ihre Programmvorschau mitzuteilen. Diese beriefen sich auf ihr Urheberrecht, gaben aber gleichzeitig eigene, entgeltlich vertrieben wöchentliche Programmvorschauen für ihre jeweiligen Programme heraus. Daraufhin erließ die Kommission eine Verbotsentscheidung, die durch EuG und EUGH bestätigt wurden. Auf die Herstellung von Interoperabilität sind folgende Gesichtspunkte, auf die der EuGH sich in seiner Entscheidung gestützt hat, von Interesse:

²⁷³ Zum ähnlich gelagerten Verfahren United States v. Microsoft, *Fleischer/Doege*, WuW, 2000, 705ff.

²⁷⁴ Zur Problematik der Wettbewerbsaufsicht in dynamischen Märkten, vgl. *Meier-Wahl/ Wrobel*, WuW 1999, 28ff.

²⁷⁵ Darstellung des Fall IBM siehe *Moritz*, CR 2004, 321 (324); instruktiv zum Begriff des funktionsfähigen Wettbewerbs, *Koenig/Kühling*, WuW 2000, 596 (597f.).

²⁷⁶ *EUGH*, Slg. 1995, I-743 –Magill; zur „essential facilities – doctrine“ im europäischen Kartellrecht, die Wettbewerbern den Zugang zu „wesentlichen Einrichtungen“ eröffnen soll, vgl. *EuGH*, GRUR Int. 1999, 262ff – Oscar Bronner; *Fleischer/Weyer*, WuW 1999, 350ff.; *Dreher*, DB 1999, 833ff.; *Schwarz van Beck*, S. 10.

- die Weigerung der Herausgabe der Informationen über die Programmvorschau stellt einen Missbrauch i.S.d. Art. 82 Abs. 2 lit. b EGV²⁷⁷ dar, weil die Fernsehanstalten zwangsläufig die einzige Quelle für die Grundinformationen über die wöchentliche Programmplanung waren,
- durch die Weigerung wurde, bei eigener Betätigung auf dem gleichen Markt „Fernsehprogrammführer“ hat diese das Auftreten eines neuen Erzeugnisses verhindert und
- das Verhalten hatte zur Folge, dass auf dem Markt „Fernsehprogrammführer“ der Wettbewerb durch die Vorenthaltung des Zugangs zu den Grundinformationen ausgeschlossen wurde.

Übertragen auf den Fall „Microsoft“ hat die Kommission in zutreffender Weise die Feststellung getroffen, Microsoft habe auf dem Markt für PC-Betriebssysteme eine Quasi-Monopolstellung²⁷⁸. Folge der Weigerung der Bereitstellung der für die Herstellung von Interoperabilität notwendigen Informationen führe zu dem Risiko des Ausschlusses von Wettbewerb auf dem Markt für Arbeitsgruppenserver-Betriebssystemen²⁷⁹, der Unterdrückung von Innovationen²⁸⁰ sowie der Einschränkung der Verbraucher in der Produktauswahl²⁸¹.

Die Entscheidung zeigt auf, dass kartellrechtliche Instrumente in der Lage sind, Marktstörungen zu beseitigen, die auf die Behinderung der Herstellung von Interoperabilität gerichtet sind zu verhindern. Einschränkung muss jedoch bemerkt werden, dass hierfür eine quasi-monopolistische Stellung des Unternehmens erforderlich ist, welche über die notwendigen Informationen verfügt. Mit Immaterialgüterrechten verbundene Monopolstellungen haben somit nicht zwingend auch ein wirtschaftliches Monopol i.S.d. Kartellrechts zur Folge. Für den Sektor von PC-Betriebssystemen mag dies gelten, nicht jedoch in dem umfassenden Sinne der erforderlich ist, um den Markt der Informationstechnik insgesamt mit den für eine Produktvielfalt erforderlichen Schnittstelleninformationen zu versorgen.

9.7 Herstellung von Interoperabilität durch Lizenzvertrag – Beispiel CIFS License Agreement, Fa. Microsoft

Wie im Rahmen der Diskussion über die Abhängigkeit der Interoperabilität aufgezeigt (S. 24), kommt der Kommunikation in Netzwerken besondere Bedeutung zu. Da sie zu einem großen Teil über Kommunikationsprotokolle erfolgt, kann das Bestehen proprietärer Kommunikationsprotokolle die Herstellung von Interoperabilität nachhaltig gefährden. In diesen Zusammenhang ist ein vom EPA der Fa. Microsoft erteiltes Verfahrenspatent über das verteilte Bearbeiten von

²⁷⁷ Zur Frage des Mißbrauchs bei Nichterteilung einer Lizenz s. auch *EuGH*, WRP 2004, 717ff. – IMS Health, unter Bezugnahme auf die *EuGH* - Entscheidungen Magill und Bronner.

²⁷⁸ Entscheidung EU-Kommission v. Microsoft, S. 118 (429), <http://europa.eu.int/comm/competition/antitrust/cases/decisions/37792/en.pdf> (letzter Abruf 24. September 2004).

²⁷⁹ Entscheidung EU-Kommission v. Microsoft, S. 207 (773), <http://europa.eu.int/comm/competition/antitrust/cases/decisions/37792/en.pdf> (letzter Abruf 24. September 2004).

²⁸⁰ Entscheidung EU-Kommission v. Microsoft, S. 207 (782), <http://europa.eu.int/comm/competition/antitrust/cases/decisions/37792/en.pdf> (letzter Abruf 24. September 2004).

²⁸¹ Entscheidung EU-Kommission v. Microsoft, S. 207 (782), <http://europa.eu.int/comm/competition/antitrust/cases/decisions/37792/en.pdf> (letzter Abruf 24. September 2004).

Dateien (Method and system for open file caching in a computer system) einzuordnen²⁸². Es handelt sich um einen proprietären Kommunikationsstandard der im Rahmen einer Rechnervernetzung zur Anwendung gelangt. Verwendet werden darf der Standard von Dritten nur dann, wenn das sog. MS CIFS License Agreement²⁸³ akzeptiert wird. In diesen Nutzungsbedingungen sind Restriktionen enthalten, die die Herstellung von Interoperabilität einzuschränken vermögen:

3.2 Patent License. *Subject to Sections 3.4 to 3.7, Microsoft hereby grants Company a worldwide, royalty-free, non-exclusive, personal, non-transferable, non-sublicensable, license under its Necessary Claims to make, use, import, offer to sell, sell and distribute, directly or indirectly, to End Users, object code versions of Company Implementations only as incorporated into Company Products and solely for the purpose of complying with the Technical Reference. The above license is limited to implementing the CIFS communication protocol itself, and **does not include** any express or implied licenses or other rights to any underlying technology (operating system technology, local file system technology, etc.) that may be used to make a complete file server or other CIFS compatible device.*

3.3 Source Code Distribution. *Company also has a worldwide, royalty-free, non-exclusive, personal, non-transferable, non-sublicensable, license to distribute or otherwise disclose source code copies of such Company Implementation licensed in Section 3.2 only if Company (i) prominently displays the following notice in all copies of such source code, and (ii) distributes or discloses the source code only under a license agreement that includes the following notice as a term of such license agreement **and does not include any other terms that are inconsistent with, or would prohibit, the following notice:***

Note "This source code may incorporate intellectual property owned by Microsoft Corporation. Our provision of this source code does not include any licenses or any other rights to you under any Microsoft intellectual property. If you would like a license from Microsoft (e.g. rebrand, redistribute), you need to contact Microsoft directly."

3.4 Limited Scope. *The above license does not include the right (express, implied or otherwise) to make, use, import, offer to sell, sell or otherwise distribute or dispose of any technology that is not explicitly described in the Technical Reference, including without limitation (a) any Microsoft CIFS implementation, (b) any Microsoft technology that embodies a Microsoft CIFS implementation, (c) any enabling technologies that may be necessary to make or use any product or portion thereof that complies with the Technical Reference, but are not themselves expressly set forth in the Technical Reference, and (d) any implementation of other specifications or technologies developed by Microsoft or others that are merely referred to in the body of the Technical Reference.*

3.5 Use of Technical Reference. *Any Company Implementation shall implement the CIFS protocol in its entirety as specified in the Technical Reference. By way of clarification of the foregoing, Company shall not be required to implement any portion of the Technical Reference that is identified in the Technical Reference as "optional". However, if Company elects to implement a portion of the Technical Reference that is identified as optional, then Company must also implement that optional portion of the Technical Reference in its entirety. Additionally, Company Implementation shall not impair the operation of any Microsoft implementation of CIFS in any Microsoft product.*

3.6 Reciprocal Patent License. *To the extent Company owns, controls or can sublicense any Necessary Claims, Microsoft is hereby granted a worldwide, royalty-free, non-exclusive, personal, non-transferable, non-sublicensable license under its Necessary Claims to make, use, import, offer to sell, sell and distribute, directly or indirectly, to End Users, fully compliant implementations of the Technical Reference solely for the purpose of complying with the Technical Reference.*

Entwickler von Anwendungssoftware, die im Regelfall eine Individualsoftware sein wird, und die Funktionalitäten z.B. in das „**Verteilte System**“ „Distributed COM“ von Microsoft einbinden müssen, haben keine andere Entscheidungsalternative als die CIFS - Lizenzbedingungen zu akzeptieren, da das CIFS ein integraler Bestandteil des Betriebssystems ist (vgl. S. 27). In den Lizenzbedingungen weist Microsoft auf zwei US-Verfahrenspatente hin (Method and system for network communications using raw mode protocols, US 5,265,261 und US 5,437,013). Aus den Patentansprüchen geht hervor, dass es sich in beiden Fällen um computerimplementierte Erfin-

²⁸² EP0438571, US Pat.No.5,261,051; <http://patent.tange.dk/ziki/EP/4/3/EP438571.html> (letzter Abruf: 23. September 2004).

²⁸³ http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cifs/protocol/royalty-free_cifs_technical_reference_license_agreement.asp (letzter Abruf 28. September 2004).

dungen handelt. Die Offenlegung der CIFS-Spezifikationen war auch Gegenstand des **EU-Kartellverfahrens** gegen die Fa. Microsoft²⁸⁴. Im Verlauf des Verfahrens kündigte Microsoft an, kostenlose Lizenzen für in dem Betriebssystem Windows NT 4.0 implementierte CIFS-Protokolle erteilen zu wollen. Hierzu dient das o.g. MS CIFS License Agreement. Benutzt werden darf das Protokoll nur dann, wenn der Lizenznehmer seinerseits gegenüber seinem Kunden die in Ziff. 3.3 des MS CIFS License Agreement aufgeführten Passus verwendet. Da die Lizenz als persönlich und nicht übertragbar ausgestaltet ist, wird der Kunde gezwungen, eine eigenständige Lizenzvereinbarung mit Microsoft abzuschließen, wenn er den Sourcecode für eigene Zwecke, z.B. im Rahmen des Redistributing – Wiederverkäufer -, benutzen möchte. Auf diesem Wege kann Microsoft direkten Zugang zu dem Kundenstamm des Lizenznehmers erlangen. Gerade in der Softwarebranche ist der Kundenstamm wesentlicher Bestandteil des Geschäfts- bzw. Firmenwertes, weshalb die Sperrwirkung, die von dieser Klausel ausgeht, nicht unterschätzt werden darf. Aus Ziff. 3.2 ergibt sich, dass an Endbenutzer erteilte Lizenzen von dieser Regelung nicht erfasst werden. Ergebnis dieser Konstruktion ist jedoch die Erlangung der Kontrolle über Systeme verteilter Softwareentwicklung, insb. Im Bereich von Open Source Produkten. Durch die Beschränkung der Weitergabe an den Endnutzer wird derjenige, der die Entwicklung weiterführen möchte gezwungen, mit Microsoft eine selbständige Lizenzvereinbarung abzuschließen. Wie die Einlassungen von Microsoft in dem aktuellen EU-Kartellverfahren zeigen, kann Reverse Engineering, wie das Beispiel „**SAMBA**“ zeigt, durchaus erfolgreich zur Herstellung von Interoperabilität genutzt werden²⁸⁵. Hierdurch wird die Bedeutung des **Reverse Engineering** für die Herstellung von Interoperabilität verdeutlicht. Im Rahmen der Bewertung der nachfolgenden Erörterungen zu der Frage des Verhältnisses von Reverse Engineering und Patenschutz wird ferner zu berücksichtigen sein, inwieweit allein die Tatsache der Möglichkeit der Analyse von Erzeugnissen oder Verfahren als Druckmittel zur Herstellung von Interoperabilität gegenüber einem nicht-offenbarungswilligen Rechteinhaber verwendet werden kann.

Zugleich muss sich der Lizenznehmer zur Einräumung von sog. **Cross-Lizenzen** für Erfindungen verpflichten, die unter Verwendung des CIFS zustande gekommen sind (Ziff. 3.6). Ein wechselseitiges Verfügbarmachen von Patentrechten ist für sich genommen nicht zu beanstanden, es ist aber nach der derzeitigen Praxis nicht ausgeschlossen, dass die Einräumung einer CIFS-Lizenz an die Bedingung der Patentierung der im Rahmen der Lizenz entwickelten Erzeugnisse oder Verfahren geknüpft wird.

Wie aufgezeigt müssen in „Verteilten Systemen“ die sog. „Schlüsselschnittstellen der Hard- und Softwaremodule offen gelegt werden. Hierzu gehört auch die Interprozesskommunikation auf Anwendungsebene. Sich allein auf eine interoperabilitätsfreundliche Lizenzpolitik des Lizenzinhabers zu verlassen, erscheint aufgrund der Erfahrungen, die mit Blick auf Geschäftspolitik der Fa. Microsoft gewonnen werden konnten, zumindest als fragwürdig.

9.8 Interoperabilität von Softwaresystemen – ausgewählte Beispiele

Im Rahmen des Kurzgutachtens werden lediglich die Beispiele des Trusted Computing, der Konvertierungsprogramme sowie der Embedded System stellvertretend für die unter 8.1 aufgeführten Softwaresysteme hinsichtlich der Rahmenbedingungen für die Herstellung von Interoperabilität untersucht. Im wesentlichen stellen sich in allen angesprochenen Fallgestaltungen ähnliche Aufgabenstellungen.

²⁸⁴ Entscheidung EU-Kommission v. Microsoft, S. 68f. (237ff.), <http://europa.eu.int/comm/competition/antitrust/cases/decisions/37792/en.pdf> (letzter Abruf 24. September 2004).

²⁸⁵ Entscheidung EU-Kommission v. Microsoft, S. 81 (292, 293.), <http://europa.eu.int/comm/competition/antitrust/cases/decisions/37792/en.pdf> (letzter Abruf 24. September 2004).

9.8.1 Trusted Computing, Nexus

Für die Herstellung von Interoperabilität mit Trusted Computing Plattformen sind Treiber und Service-Provider-Schnittstellen erforderlich, die über den sog. TCG Software Stack (TSS) erfolgen. Wie unter 6.5.1 dargestellt, ist die Offenlegung der Schnittstellen erforderlich.

Die Fa. IBM hat für Open Source Anwendungen die unterste Treiberebene ihres TSS-Stacks als Open Source Linux Implementierung offen gelegt²⁸⁶.

Entscheidend für die Sicherstellung der Interoperabilität wird insofern die Lizenzpolitik der die Schutzrechte innehabenden Unternehmen sein. Die Vorgehensweise der Fa. IBM weicht, im Fall des TSS-Stacks von der der Fa. Microsoft im Fall CIFS-Protokoll (s. 8.3.3.2) in nicht unerheblicher Weise ab. Die Lizenzbedingungen der Fa. IBM basieren auf Urheberrechtsschutz und sind wie folgt formuliert:

(Auszug)

“Copyright (c) 2003, IBM. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

** Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.*

** Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.*

** Neither the name of IBM nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.”*

Zum einen fällt auf, dass offenbar keine Bemühungen unternommen wurden Patentschutz, z.B. im Rahmen einer computerimplementierten Erfindung zu erlangen, und zum anderen, dass keine Beschränkung der Nutzungsrechte im Fall des Wiederverkauf vorgesehen wird. So ist zwar ein Urheberrechtshinweis anzubringen, der Nutzer wird aber nicht gezwungen, eine unmittelbare Lizenzvereinbarung mit der Fa. IBM abzuschließen. Die Praxis ist jedoch, wie aufgezeigt (vgl. 5.5.1), uneinheitlich.

Die durch die TCG veröffentlichten Spezifikationen sollen den außen stehenden dritten Entwickler in die Lage versetzen, unmittelbar über die hierin enthaltenen technischen Spezifikationen und Protokolle auf die Schnittstellen zugreifen zu können, über die TC-Bausteine – Chips – angesprochen werden. Der freie Nutzung von Schnittstelleninformationen, dient der Herstellung von Interoperabilität. Erstellt wurden die Spezifikationen durch Mitglieder der TCG, so dass diese gemeinsam die entstehenden Schutzrechte innehaben. Aus diesem Grund sieht Sec. 16.9 TCG - Satzung die Einräumung einer einfachen Urheberrechts-Lizenz vor, die Spezifikationen zur Zwecken der Entwicklung und Veröffentlichung u.a. vervielfältigen und verbreiten zu können. Entsprechend sind die Urheberrechtsvermerke z.B. auf der TCPA – Main Specification oder der TCG Specification Architecture Overview ausgestaltet worden. Danach dürfen die Spezifikationen nur für den internen Gebrauch des Entwicklers verwendet werden²⁸⁷. Hierdurch wird dem Zweck genüge getan, Interoperabilität herzustellen. Ist eine über den Zweck der Entwicklung hinausgehende Nutzung beabsichtigt, muss mit dem unmittelbaren Rechteinhaber eine Lizenzvereinbarung abgeschlossen werden. In gewissem Umfang einschränkend wirkt lediglich Sec. 16.2 (b) TCG – Satzung, die den Mitgliedern der TCG vor der Veröffentlichung einer Spezifikation für eine Zeitspanne von 60 Tagen ein Prüfungsrecht einräumt. Mitgliedern der TCG

²⁸⁶ <http://www.research.ibm.com/gsal/tcpa/tpm-1.1b.tar.gz> (letzter Abruf 4. Oktober 2004).

²⁸⁷ „No license, express or implied, by estoppel or otherwise, to any TCG oder TCG member intellectual property rights is granted herein. Except that a license is hereby granted by TCG to copy and reproduce this specification for internal use only.”

wird mithin ein zeitlicher Vorsprung, was die Entwicklung von auf der Spezifikation aufbauender Entwicklungen verschafft.

Standardisierung und Normung können somit als ein die Interoperabilität förderndes, wenn nicht sicherstellendes Instrument wirken. Standards, die z.B. innerhalb der TCG entwickelt werden, sind nach deren Satzung innerhalb der Organisation gegenüber den jeweils interessierten Mitgliedsunternehmen nicht nur zur satzungsgemäßen Verwendung offen zulegen²⁸⁸, sondern die TCG selbst ist berechtigt, die Standards, so etwa als Downloads auf der Homepage der TCG, zu verbreiten. Über eine Lizenzierung von Rechten gegenüber Dritten, die über die Entwicklungstätigkeit hinausgehen entscheidet jedoch der Rechteinhaber unabhängig hiervon.

Weitaus größere Bedeutung kommt der konsensbildenden Wirkung von Standardisierungsprozessen im Bereich der Informationstechnologie zu. So sehen etwa die ETSI IPR Policy-Regelungen²⁸⁹ im Fall des Nichtbestehens von Lizenzbereitschaft eines Patentinhabers die Überarbeitung des Standards vor, damit die entsprechende technische Spezifikation nicht durch ein bestehendes Schutzrecht blockiert werden kann²⁹⁰. Gleiches gilt für ISO- oder DIN-Normungsverfahren. Ist im Rahmen einer Normung die Bezugnahme auf ein Patent nicht vermeidbar, so muss sich der Patentinhaber verpflichten, die Nutzung seines Patentbesitzes allen Interessierten zu fairen und nicht diskriminierenden Bedingungen einzuräumen. Ist das nicht gewährleistet, so wird die Norm im Regelfall nicht verabschiedet (siehe 6.4). Standardisierungs- bzw. Normungsprozesse können in diesen Fällen nur dann erfolgreich abgeschlossen werden, wenn die betreffenden Patentinhaber ein Interesse an der Verabschiedung der entsprechenden Standards oder Normen haben. Marktmachtstarke Patentinhaber verfügen deshalb über weit reichende Einflussmöglichkeiten auf die Entwicklung von Märkten, indem sie die fördernde Wirkung von Standards verhindern können.

Einschränkend muss jedoch aus der Sicht des Patentrechts festgestellt werden, dass sich aus derartigen Standardisierungs- und Normungsaktivitäten keine weitergehenden Auswirkungen ergeben. Selbst wenn die entsprechenden Informationen Teil einer ISO- oder DIN-Norm sein sollten, wird hierdurch nicht der Eingriff in fremde Schutzrechte gerechtfertigt, ebenso wenig, wie die Mitwirkung des Patentinhabers an der Erstellung der Norm den Rückschluss zulässt, er habe zumindest stillschweigend eine Einwilligung in die Benutzung des Patents erteilt²⁹¹. In die Eigentumsrechte, hier das wie ein Monopol wirkende Recht Dritte von der Benutzung des Patents durch Nichtlizenzierung ausschließen zu können, greift somit weder die Standardisierung, noch die Normung ein.

Da Entwicklungen auf dem Gebiet des „Trusted Computing“, ebenso wie in anderen Bereichen der DV-Technik, als Teil einer computerimplementierten Erfindung in Verbindung mit einem dem Patentschutz zugänglichen Erzeugnis²⁹² oder Verfahren²⁹³ Patentschutz erlangen können, sind wegen der Offenlegungswirkung von Standards und Normen keine negativen Auswirkungen auf die Herstellung von Interoperabilität gegeben.

²⁸⁸ Sec. 15f. Bylaws of Trusted Computing Group, https://www.trustedcomputinggroup.org/about/tcg_bylaws.pdf (letzter Abruf 23. September 2004)

²⁸⁹ Zur Entwicklung der Normungspolitik der ETSI, *Kübel*, S. 56ff.

²⁹⁰ Ziff. 8.1 ETSI IPR Policy-Regelungen, http://www.etsi.org/legal/documents/ETSI_IPRPolicy.pdf (letzter Abruf 4. Oktober 2004).

²⁹¹ *Keuckenschrijver* in: Busse Patentgesetz § 11 Rndnr. 28 m.w.Nw.

²⁹² Z.B. Chip, Motherboard o.ä.

²⁹³ Hier ist die sich US-amerikanischer Praxis annähernde Patentierungspraxis des EPA zu berücksichtigen.

9.8.2 Konvertierungsprogramm/Nachrichten-Broker

Voraussetzung für die Erfüllung dieser Aufgabe ist die Durchführung der gleichen Entwicklungsschritte, die für einen unmittelbaren Zugriff auf eine Softwareschnittstelle sind. Mittels Reverse Engineering muss die Art und Weise ermittelt werden, in der das Betriebssystem die Kommunikation zu Anwendungen außerhalb des Betriebssystems herstellt.

Nachrichten-Broker die die Funktion eines Gateway übernehmen, sind Rechner (Hardware) mit entsprechender Software, welche der Nachrichtenübermittlung dienen. Sie werden daher auch als Kommunikationsserver bezeichnet (s. 6.1.2). Da die Voraussetzungen für die Herstellung von Interoperabilität die gleichen sind wie im Fall der Konvertierungsprogramme, ergeben sich die gleichen patentrechtlichen Folgen.

Auch in den Fällen Konvertierungsprogramm und Nachrichten-Broker übt somit das Patentrecht, wegen der Einordnung als Neuherstellung einer dem Patentschutz unterliegenden computerimplementierten Erfindung, eine Sperrwirkung aus. Notwendige Voraussetzung wäre demnach auch hier ein entsprechendes Benutzungsrecht des Herstellers des Konvertierungsprogramms bzw. des Nachrichten-Brokers.

9.8.3 Embedded Systems

Fragen der Interoperabilität stellen sich dann, wenn mehrere Ein-Chip-Computersysteme in ein Gesamtsystem eingegliedert werden sollen. Voraussetzung für die Funktionsfähigkeit des Gesamtsystems ist die Kommunikationsfähigkeit der Teilsysteme. Die Anforderungen an die Interoperabilität sind mit denjenigen vergleichbar, die auch bei „normalen“ Softwaresystemen ergeben. Es handelt sich somit um klassische Fälle computerimplementierter Erfindungen.

Besonderheiten ergeben sich hinsichtlich der praktischen Durchführbarkeit eines Reverse Engineering, das unabhängig vom Hersteller zu Zwecken der Interoperabilität erfolgen soll. Unterschieden wird zwischen offenen und geschlossenen Systemen. Letztere verhindern nahezu vollständig ein Auslesen des Chips und damit den für ein Reverse Engineering notwendigen Zugriff auf die implementierte Software.

Voraussetzung für den interoperablen Einsatz von „Embedded Systems“ ist folglich die Offenlegung der Schnittstellen, über die auf die im Chip implementierte Software zugegriffen werden kann. Als computerimplementierte Erfindung ist ein Embedded System dem Patentschutz zugänglich. Ein gewerblichen Zwecken dienendes Reverse Engineering ist mithin, wie dargestellt, keine erlaubte Benutzung und führt, wegen Verstoß gegen § 9 PatG, zu einer Patentverletzung. Soll auch unter dem Regime des Patentrechts Herstellern von Software die Möglichkeit der Herstellung von Interoperabilität mit Embedded Systems eröffnet werden, muss zwingend die Offenlegung der Schnittstellenbeschreibung erfolgen. Wegen der zumindest zum Teil schon aus technischen Gründen nicht bestehenden Möglichkeit der Durchführung eines Reverse Engineering ergibt sich hier ein Sonderfall, da die interoperabilitätsrelevanten Informationen nicht auf anderem Wege erlangt werden können. Sichergestellt werden kann diese letztlich nur dann, wenn in Rahmen der Patentanmeldung eines Embedded System in der Patentschrift die Schnittstellinformationen im Rahmen der Offenbarung der Erfindung nach § 34 Abs. 4 PatG zwingend offen gelegt würden.

9.9 Zusammenfassung – Interoperabilität und PatG

Im Kapitel 6 wurde u.a. erläutert, dass zwischen Hard- und Softwareschnittstellen unterschieden wird. Lediglich die letztgenannte Variante war im Rahmen des Gutachtauftrages im Weiteren zu untersuchen.

Einer Softwareschnittstelle im Rechnersystem kommt die Aufgabe zu, Informationen zu transportieren, so etwa von einem Betriebssystem zu einer Applikation. Das wie, d.h. die Art und Weise der Informationsbeförderung wird bei einer Softwareschnittstelle durch eine entsprechende Programmierung festgelegt, die Teil der Software insgesamt ist. Aus informationstechnischer Sicht ist die Umsetzung der Softwareschnittstelle integraler Bestandteil der Gesamtlösung, mit der Folge, dass diese keine eigenständige Bedeutung zu erlangen vermag. Aus diesem Grund ist, nach dem Kenntnisstand der Verfasser, bis dato in Europa auch noch kein Versuch unternommen worden, für eine Softwareschnittstelle als solche ein Patent zu erlangen, zumal in einem solchen Fall der Patentierungsausschluss für Computerprogramme als solche nach § 1 Abs. 2 Nr. 3 PatG, Art. 52 Abs. 2 lit. c EPÜ zu beachten ist. Softwarebezogene Schnittstellen können folglich, ebenso wie Dateiformate und andere nicht selbständig patentfähige Entwicklungen, nur als Teil einer computerimplementierten Erfindung Patentschutz erlangen.

Softwareschnittstellen „als solche“ sind dem Patentschutz nicht zugänglich, sondern können nur als Teile eines Erzeugnisses oder Verfahrens an der Schutzwirkung des Patents teilhaben. Wie aufgezeigt, ist es zweifelhaft, ob das Patentrecht de lege lata Maßnahmen zur Herstellung von Interoperabilität zulässt (vgl. Diskussion zu § 11 Nr. 1 und 2 PatG). Nach hier vertretener Auffassung ist Herstellern von Software damit die Möglichkeit abgeschnitten, im Wege des Reverse Engineering vom Hersteller nicht freiwillig bekannte Schnittstelleninformationen erlangen zu können. Folglich verbleibt es bei dem patentrechtlich gewährten Ausschließlichkeitsrecht des Patentinhabers.

Wegen der kurzen Entwicklungszyklen, durch die die Softwareerstellung in besonderer Weise gekennzeichnet ist, können weder Zwangslizenzerteilungsverfahren noch kartellrechtliche Begrenzungen der Marktmacht von marktbeherrschenden Patentinhabern einem möglichen Missbrauch des Monopolrecht zeitnah entgegenwirken. Nun mag man einwenden, dass dem auf die Schnittstelleninformationen angewiesenen Softwarehersteller die Möglichkeiten des einstweiligen Rechtsschutzes offen stehen, um eine schnelle, wenn auch nur vorläufige Entscheidung herbeiführen zu können. Da es sich bei der Mehrzahl der betroffenen, auf Schnittstelleninformationen angewiesenen Unternehmen um KMU's handeln wird, wie die Analyse der Marktstruktur der IT-Wirtschaft ergeben hat (siehe 3.1), werden diese Möglichkeiten infolge des nach wie vor bestehenden hohen wirtschaftlichen Risikos im Fall des negativen Ausgangs des Hauptsacheverfahrens nicht wahrgenommen. Zu berücksichtigen ist ferner, dass die überwiegende Zahl mittelständischer Softwareersteller nicht über die finanziellen Spielräume verfügt, die genannten Instrumente einsetzen zu können. Somit sind diese in der Praxis zwar theoretisch einsetzbar, aber im Ergebnis praktisch wirkungslos.

Hieraus folgt, dass der auf die Schnittstelleninformationen angewiesene Dritte durch das Nadelöhr „Lizenzierungspolitik des Schutzrechtsinhabers“ hindurchgehen muss. Je nach aktueller Lage kann sich diese verändern. Damit fehlt es an der notwendigen Rechts- und Planungssicherheit. Auf diesem Wege kann der Schutzrechtsinhaber Entwicklungsprozesse von Wettbewerbern oder dritten Softwareherstellern maßgeblich beeinflussen, wenn nicht sogar steuern. Nicht zuletzt ist die Europäische Kommission im Fall Microsoft von „quasi-monopolistischen“ Strukturen ausgegangen, die letztlich eine marktbeherrschende Stellung entstehen lassen²⁹⁴. Da im Bereich von Lizenzvereinbarungen in Grundsatz uneingeschränkte Vertragsfreiheit besteht, kann durch Nichterteilung oder verzögerte Erteilung einer Lizenz bzw. durch für den Li-

²⁹⁴ Hier ist auf einen weiteren Unsicherheitsfaktor hinzuweisen, nämlich dem der kartellrechtlichen Feststellung einer mißbräuchlichen Ausnutzung einer marktbeherrschenden Stellung des Schutzrechtsinhabers. Es muss die „Funktionsfähigkeit des Wettbewerbs“ gewährleistet sein, wobei die Auslegung des Begriffs stark umstritten ist; vgl. *Koenig/Kühling*, WuW 2000, 596 (598ff.). Besonders deutlich wurde die Problematik in der EuGH Entscheidung IMS Health (WRP 2004, 717ff.), insbesondere im Hinblick auf die Zumutbarkeit der Zusammenarbeit mit anderen Wirtschaftsteilnehmern und dem Ausweichen auf Alternativprodukte.

zennnehmer besonders belastende Lizenzbedingungen de facto die Lizenzerteilung blockiert werden. An Hand des Beispiels der CIFS-Lizenzbedingungen konnte aufgezeigt werden, dass der Lizenznehmer gezwungen werden kann, Informationen preiszugeben, die einen wesentlichen Bestandteil des Geschäfts- bzw. Firmenwertes darstellen. Berücksichtigen muss man ferner die Tatsache, dass KMU's im Regelfall nicht über eigene Patente verfügen und es ihnen in der Folge an Verhandlungsmasse fehlt, um etwa in Cross-Lizenzvereinbarungen mit Patentinhabern eintreten zu können. Mit dem Anwachsen der Zahl von Patenten, deren Gegenstand computerimplementierte Erfindungen sind und der Patentkonzentration in sog. „Patent-Pools“, entsteht für KMU's zunehmend ein „Schutzrechtsflaschenhals“, der die Entwicklung von Softwareprodukten zu einer Haftungsfalle werden lässt, da diese meist nicht über die sachlichen oder personellen Ressourcen verfügen, um mögliche Schutzrechtsverletzungen rechtzeitig erkennen zu können. Zugleich wird mit der wachsenden Zahl von Patenten das „Um-ein-Patent-Herumprogrammieren“ immer schwieriger, wenn nicht unmöglich.

Die Sperrwirkung solcher Vertragsbedingungen darf daher nicht unterschätzt werden, da die Lizenzvereinbarung unter den geschilderten Rahmenbedingungen den einzigen Ausweg aus der „Haftungsfalle Patentverletzung“ darstellen wird. Ferner ist fraglich, ob derartige Anforderung diskriminierende Wirkung haben bzw. als angemessen und geschäftsüblich qualifiziert werden können und wird letztlich einzelfallbezogen zu beantworten sein. Inwieweit die Regelungen des Kartellrechts sowie die Möglichkeit der Erteilung von Zwangslizenzen hier Abhilfe schaffen können, kann nach derzeitigen Erkenntnisstand nicht mit hinreichender Sicherheit beantwortet werden, da es an der hierfür erforderlichen Dichte von Rechtsprechung mangelt. Nach hier vertretener Ansicht sind diese aufgrund ihrer Struktur zu schwerfällig und bergen, wegen der regelmäßig hohen Streitwerte, Risiken in sich, die ein mittelständisches Unternehmen, und schon gar nicht ein Einzelkaufmann, zu tragen nicht in der Lage ist.

Da auch befristet gewährte Lizenzen erteilt werden können, muss weiter der Faktor „Unsicherheit“ in die Bewertung einfließen. Schutzrechteinhaber können bei entsprechender Vertragsgestaltung sich Optionen für u.U. kurzfristige Veränderungen der Lizenzpolitik schaffen. Im Zweifel wird der Lizenznehmer ein auf seiner Seite bestehendes Interesse an einer langfristig erteilten Lizenz wegen fehlender Marktmacht, nicht durchsetzen können. Gleiches gilt für die Vergütungsmodelle. Erfahrungsgemäß erweisen sich umsatzabhängige Vergütungsregelungen für Mittelständler als besonders problematisch, da diese ohnehin mit einer niedrigen Umsatzrentabilität vor Steuern arbeiten. Nicht selten werden KMU's schon heute gezwungen erhebliche Summen für die Abwehr unberechtigt erhobener Forderungen aufzuwenden. Nach eigenen Erhebungen müssen KMU's z.T. Kosten in einer Höhe von deutlich mehr als 200.000€/p.a. zur Verteidigung in marken- und patentrechtlichen Streitigkeiten aufwenden. Hierdurch sinkt der Spielraum für Investitionen und gefährdet damit die Bestandssicherung der betroffenen Unternehmen.

Soweit die Lizenzbedingungen als Allgemeine Geschäftsbedingungen verwendet werden, kommt die Kontrolle nach den §§ 305ff. BGB in Betracht. Da diese im Zweifel vom Lizenzgeber gegenüber einem Unternehmer nach §§ 310 Abs. 1 Satz 1, 14 BGB verwendet werden, findet lediglich eine reduzierte Prüfung der Unangemessenheit der Regelungen nach § 307 Abs. 1 und 2 BGB statt. Dabei ist im Rahmen der Prüfung einer möglichen unangemessenen Benachteiligung, wie etwa im Fall des Abweichens von wesentlichen Grundgedanken einschlägiger gesetzlicher Regelungen oder der Einschränkung vertragswesentlicher Rechte oder Pflichten auf die im Handelsverkehr geltenden Gewohnheiten und Gebräuche angemessen Rücksicht zu nehmen. Ob sich entsprechende Gebräuche bereits herausgebildet haben, ist noch nicht untersucht worden. Ein Handelsbrauch gibt die Verkehrssitte des Handels wieder und wird definiert als verpflichtende Regel, die auf einer gleichmäßigen, einheitlichen und freiwilligen Übung der beteiligten Verkehrskreise für vergleichbare Geschäftsvorfälle beruht, die sich über einen längeren Zeitraum, auf der Grundlage einer einheitlichen Auffassung der Beteiligten, gebildet hat²⁹⁵.

²⁹⁵ BGH NJW 1994, 659.

Bereits aufgrund der Tatsache, dass erst in jüngerer Zeit Lizenzbedingungen in Rahmen der Informationsbereitstellung zur Herstellung von Interoperabilität verwendet werden, kann sich kein entsprechender Handelsbrauch herausgebildet haben. Es verbleibt mithin allein die Möglichkeit über die Feststellung der Unangemessenheit der Vertragsbedingungen deren Unwirksamkeit herbeizuführen. Die Unangemessenheitskontrolle richtet sich am Leitbildcharakter eines patentrechtlichen Lizenzvertrages aus, weshalb zu berücksichtigen ist, dass der Patentinhaber frei über die Einräumung von Benutzungsrechten entscheiden kann. Folglich ist er insoweit allenfalls den sich aus §§ 134, 138 BGB ergebenden Einschränkungen unterworfen. Aber selbst im Fall der Annahme eines Verstoßes gegen AGB-rechtliche Grundsätze kann nicht in jedem Fall von der Korrekturmöglichkeit durch die §§ 305ff. BGB ausgegangen werden, da hierfür Voraussetzung ist, dass die Anwendbarkeit deutschen Rechts gegeben ist. Auch hier zeigt der Fall CIFS-Lizenzbedingungen der Fa. Microsoft, dass durch entsprechende, im Handelsverkehr zulässige Rechtswahl- und Gerichtsstandsvereinbarungen (Art. 27 EGBGB, §§ 38f. ZPO), problematische AGB der Beurteilung und damit der Korrektur durch z.B. deutsches AGB-Recht, entzogen werden können.

Im Ergebnis stellt weder das deutsche Patentrecht noch das EPÜ ein rechtliches Instrumentarium zur Verfügung, das in der Lage ist, derartige Blockaden unter branchennotwendigen Randbedingungen (Zeit und Kosten) zu verhindern. Wie das Beispiel Microsoft zeigt, wird nichts unversucht gelassen, die Offenlegung von Schnittstelleninformationen im Einzelfall hinauszuzögern oder gar zu verhindern. So hat Microsoft nunmehr in dem noch laufenden Kartellverfahren den Versuch unternommen, Schnittstelleninformationen als Geschäftsgeheimnisse zu deklarieren und behauptet, dass hieraus Microsoft nicht hinzunehmende Gefährdungen erwachsen würde²⁹⁶.

10 Meilensteine des Richtlinienentwurfs

In dem folgenden Kapitel wird der Schwerpunkt der Betrachtung auf die Diskussion der Herstellung von **Interoperabilität** im Rahmen des Behandlung des Richtlinienentwurfs gelegt. Nur soweit für das Verständnis unbedingt notwendig, wird auf in dem Richtlinienentwurf enthaltene Patentverteilungsvoraussetzungen eingegangen.

Computerprogramme „als solche“ können de lege lata weder nach dem EPÜ, noch den Patentgesetzen der Mitgliedsstaaten patentiert werden. Die Tatsache, dass in erheblichem Umfang Patente für computerimplementierte Erfindungen Patentschutz erteilt wurden, hat zu Rechtsunsicherheiten geführt, die die Kommission zum Anlass für die Vorlage des Entwurfs der Richtlinie „computerimplementierte Erfindungen“ genommen hat. Zahlreiche Patente dieser Gruppe stehen in enger Verbindung zu Kernbereichen der Informationstechnik (digitale Bildverarbeitung, Datenerkennung, -darstellung und Speicherung sowie für programmgesteuerte Prozessoren in den Bereichen Kraftfahrzeugtechnik und Maschinenbau)²⁹⁷. Algorithmen als solche scheiden als Gegenstand eines Patents somit nur dann aus, wenn er ohne Bezug zu einer physischen Umgebung definiert wird²⁹⁸.

Die politische Diskussion zu dem Richtlinien-Entwurf der Europäischen Kommission und des Rates verläuft äußerst kontrovers und wird zunehmend emotional geführt²⁹⁹. Im Zentrum der

²⁹⁶ <http://de.internet.com/dynamic/print.php?id=2031233>.

²⁹⁷ Vorschlag für eine Richtlinie des Europäischen Parlaments und des Rates über die Patentierbarkeit computerimplementierter Erfindungen, KOM (2002) 92 endg., S. 2.

²⁹⁸ Vorschlag für eine Richtlinie des Europäischen Parlaments und des Rates über die Patentierbarkeit computerimplementierter Erfindungen, KOM (2002) 92 endg., S. 8.

²⁹⁹ Vgl. Bericht über eine beim DPMA geführte Podiumsdiskussion zum Richtlinien-Entwurf, <http://www.heise.de/newsticker/meldungen/52425> (letzter Abruf 25. Oktober 2004).

Diskussion steht die Frage der Gewährung von Patentschutz für computerimplementierte Erfindungen, d.h. die Voraussetzungen, die erfüllt sein müssen, wenn ein Patent erteilt werden soll³⁰⁰. Der Gesichtspunkt der Interoperabilität wird in dieser Diskussion eine eher untergeordnete Rolle, wobei offenbar die Vorstellung herrscht, bereits die engere Fassung des Begriffs „technischer Beitrag“, könne die Erteilung von sog. „Trivialpatenten“ verhindern und damit zugleich die Gewährleistung von Interoperabilität sicherstellen³⁰¹.

10.1 Grundannahmen des Richtlinienentwurfes

Einer der primären Zwecke des Richtlinienentwurfes ist die Festschreibung der Patenterteilungspraxis des EPA sowie dessen Spruchpraxis der Beschwerdekammern hinsichtlich der Behandlung computerimplementierter Erfindungen. Diese Lücke hat sich nicht zuletzt als Folge des Scheitern der Bemühungen um die Verabschiedung der Verordnung über das Gemeinschaftspatent ergeben. Deutlich wird dieser Ansatz durch die Feststellung, dass die Richtlinie den Schutz computerimplementierter Erfindungen ohne abrupte Änderung der Rechtslage harmonisieren soll und dabei zugleich keine Ausdehnung der Patentierbarkeit auf Computerprogramme als solche und computerimplementierte Geschäftsmethoden vorzunehmen³⁰². Aus diesem Grunde wird nicht auf das Kriterium des technischen Beitrags verzichtet.

Die Voraussetzungen, unter denen Patente für computerimplementierte Erfindungen, in der Fassung des Entwurfs des Ministerrates vom 24. Mai. 2004, erteilt werden können, schreiben unverändert die derzeit bestehende Praxis fest. Eine Ausweitung des Patentschutzes auf reine Geschäftsmethoden wird nunmehr durch die Aufnahme des Patentierungserfordernisses des „technischen Beitrags“ verhindert. Auf eine Eingehende Analyse der Patentierungsvoraussetzungen von computerimplementierten Erfindungen wird an dieser Stelle verzichtet, da der Untersuchungsgegenstand des Gutachtens auf die Frage der Herstellung von Interoperabilität gerichtet ist³⁰³. Da die Rechtslage nach der derzeitigen Fassung des Richtlinien-Entwurfs (24. Mai. 2004) auf der Grundlage der aktuellen Praxis festgeschrieben, besteht insofern in diesem Gutachten kein Untersuchungsbedarf. Die in Kapitel 8 erzielten Ergebnisse werden durch den Richtlinien-Entwurf somit nicht verändert.

Die sich als Folge des Richtlinienentwurfes ergebende Kollision mit den aus der Richtlinie über den Rechtsschutz von Computerprogrammen ergebenden Rechte des Lizenznehmers wird mit dem Hinweis begegnet, dass bei missbräuchlicher Verwendung von Patenten auf die Möglichkeit der Erteilung von Zwangslizenzen oder Regelungen des Wettbewerbsrechts zurückgegriffen werden könne. Wie der Begriff der „missbräuchlichen“ Verwendung in der Praxis zu handhaben ist, wird nicht erläutert. Auch die Wirksamkeit der genannten Handlungsalternativen wird nicht aufgezeigt. Zudem wird auf die Ergänzungsfunktion von Urheber- und Patentrecht verwiesen, ohne auf den unterschiedlichen Charakter, Schutzzweck und Regelungssystematik der angesprochenen Schutzrechtssysteme einzugehen. Die Wirkungskraft des Art. 6 des Richtlinienentwurfes bleibt deshalb unklar.

³⁰⁰ Siehe auch jüngste Initiativen der Bundestagsfraktionen FDP (BT-Drs. 15/3240), CDU/CSU-Fraktion im Deutschen Bundestag (BT-Drs. 15/3941) und SPD/Bündnis 90 – Die Grünen (BT-Drs. 15/4034).

³⁰¹ Vgl. BT-Drs. 15/3941, S. 2.

³⁰² Vorschlag für eine Richtlinie des Europäischen Parlaments und des Rates über die Patentierbarkeit computerimplementierter Erfindungen, KOM (2002) 92 endg., S. 12.

³⁰³ Eine ausführliche Analyse der Patentierungsvoraussetzungen in den unterschiedlichen Fassungen des Vorschlag für eine Richtlinie des Europäischen Parlaments und des Rates über die Patentierbarkeit computerimplementierter Erfindungen geben *Sedlmaier/Gigerich*, S. 23ff. auf den an dieser Stelle inhaltlich verwiesen wird.

10.2 Übertragbarkeit urheberrechtlicher Grundsätze auf das Patentrecht?

Als problematisch erweist sich die Abgrenzung der Schutzrechtssysteme Urheberrecht und Patentrecht im Fall vom Computerprogrammen, da sie in beiden Systemen als Regelungsgegenstand angesprochen werden³¹³. *Busche* spricht insofern von einer „eigenartigen Zwitterstellung“³¹⁴. Abgrenzungsmerkmal ist das der Technizität. Das Urheberrecht gewährt dem Urheber Schutz für Werke der Literatur, Wissenschaft und Kunst, sofern es sich um eine persönliche geistige Schöpfung handelt (§§ 1, 2 Abs. 2 UrhG). Er weist für den Fall des Softwareschutzes auf die Tatsache der Zuordnungsentscheidung des Gesetzgebers hin, die den Softwareschutz dem Urheberrecht zugeordnet hat und im Patentrecht den Ausschluss von der Patentierbarkeit für Computerprogramme „als solche“ vorsieht und ihnen somit nicht der Charakter einer Erfindung beigemessen werden kann (§ 1 Abs. 2 PatG, Art. 52 Abs. 2, 3 EPÜ)³¹⁵.

Nicht nur die Gegenstände des Schutzes sind hinsichtlich Charakter und Funktion verschieden, sondern auch die Schutzrechtssysteme selbst. So erfasst nur das Patentrecht neue Regeln zur Erzielung eines bestimmten Erfolgs mittels beherrschbarer Naturkräfte. Wenn auch die Tätigkeit des Erfinders schöpferischer Natur ist, d.h. aus geistiger Schöpfung resultiert, geht es um Technik die das Patentrecht schützt und nicht um ästhetisch, wissenschaftlich oder literarische Werke, die dem Urheberschutz zugewiesen sind. Zum Ausdruck kommt die besondere Nähe zwischen Urheber und Werk durch das Urheberpersönlichkeitsrecht (§§ 12 – 14 UrhG), welches dem Patentrecht unbekannt ist. Auf der anderen Seite kommt dem Urheberrecht hinsichtlich der Nutzung des Urheberrechts keine dem Patentrecht vergleichbare **Sperwirkung** zu. Urheberrechtliche Verwertungsrechte erfahren vielfältige Schranken, die den Urheber Duldungspflichten in Bezug auf die Nutzung seines Werkes durch Dritte auferlegen, so etwa die Bearbeitung nach § 3 UrhG³¹⁶. Wenn auch Computerprogramme innerhalb des Urheberrechts eine Sonderbehandlung erfahren, indem die Schwelle für die Schutzwürdigkeit in den §§ 69a – 69g UrhG abgesenkt wurde – Schutz der „kleinen Münze“. Schwierigkeiten bezüglich der systematischen Einordnung der computerprogrammspezifischen Regelungen ergeben sich zumindest in Teilbereichen aus der Tatsache, dass der Gesetzgeber einige Regelungen der EG-Richtlinie über den Rechtsschutz von Computerprogrammen vom 14. Mai 1991³¹⁷ inhaltlich unverändert umgesetzt hat. Wegen der mitunter komplizierten Formulierungen ist diese Vorgehensweise nachvollziehbar, so etwa für das Dekompilierungsrecht –Reverse Engineering. Die Regelung in

³¹³ *Kraßer*, Patentrecht, S. 23.

³¹⁴ *Busche*, Patentsschutz für computerimplementierte Erfindungen 2003, S: 4, <http://www.jura.uni-duesseldorf.de/dozenten/busche/Vortragmat/softwarevortragEPA.pdf> (letzter Abruf 17. September 2004).

³¹⁵ *Busche*, Patentsschutz für computerimplementierte Erfindungen 2003, S. 5f.

³¹⁶ Z.B. die Übersetzung eines Werkes.

³¹⁷ ABL. EG Nr. 1122/42.

§ 69e UrhG entspricht der des Art. 6 der Computerprogramm-Richtlinie³¹⁸. Sie ist danach nur zulässig, wenn die zum Zweck der Herstellung der Interoperabilität eines unabhängig geschaffenen Computerprogramms mit anderen Programmen dient und die Handlungen vom Lizenznehmer oder anderen zur Verwendung einer Programmkopie berechtigten Personen vorgenommen werden³¹⁹. Im Einzelfall kann diese dazu führen, dass auch nur ein Programmteil einem Reverse-Engineering-Prozeß unterzogen werden darf, wenn in diesem Programmteil für sich genommen die für die Interoperabilität erforderliche Informationen bereitliegen³²⁰. Dekompilierung zu anderen Zwecken ist dagegen in Gänze verboten, d.h. auch im wissenschaftlichen oder privaten Bereich. *Marly* sieht hierin einen Widerspruch zu dem urheberrechtlichen Grundsatz, dass die Forschungstätigkeit nicht behindert und rein private Interessen nicht beschränkt werden sollen³²¹, ebenso wie durchaus berechtigte Interessen des Anwenders an der Programmpflege, selbst bei der Herstellerinsolvenz, sowie Interessen von Serviceunternehmen, die nicht mit dem Hersteller identisch sind, keine Berücksichtigung finden³²². Im Rahmen der Umsetzung der Computerprogramm-Richtlinie mussten daher entsprechende Systembrüche hingenommen werden, da ein Ausgleich der unterschiedlichen Interessenlagen von Herstellern und Nutzern sowie der Erhaltung des freien Wettbewerbs im Ergebnis zu einer komplizierten Kompromisslösung geführt hat.

Infolge der unterschiedlichen Konzeption der Schutzrechte Urheber – und Patentrecht sind im Falle der Anwendbarkeit beider Rechte auf einen einheitlichen Lebenssachverhalt jeweils eine gesonderte Prüfung vorzunehmen. Sie können sich somit auch nicht gegenseitig substituieren³²³. Somit ist es auch nicht möglich, eine in dem einen Schutzrechtssystem bestehende Lücke durch analoge Anwendung eines Grundsatzes zu schließen, der dem anderen Schutzrechtssystem zugeordnet ist. Dies gilt selbst dann, wenn es sich um einen Regelungsgegenstand handelt, für den beide Schutzrechtssysteme Regelungen vorsehen.

Im Ergebnis stellt weder das deutsche Patentrecht noch das EPÜ ein rechtliches Instrumentarium zur Verfügung, das in der Lage ist, derartige Blockaden unter branchennotwendigen Randbedingungen (Zeit und Kosten) zu verhindern. Wie das Beispiel Microsoft zeigt, wird nichts unversucht gelassen, die Offenlegung von Schnittstelleninformationen im Einzelfall hinauszuzögern oder gar zu verhindern. So hat Microsoft nunmehr in dem noch laufenden Kartellverfahren den Versuch unternommen, Schnittstelleninformationen als Geschäftsgeheimnisse zu deklarieren und behauptet, dass hieraus Microsoft nicht hinzunehmende Gefährdungen erwachsen würden³²⁴.

10.3 Die Diskussion des Richtlinienentwurfs unter dem Blickwinkel „Interoperabilität“

Die folgende Übersicht beschränkt sich auf eine zeitlich geordnete tabellarische Auflistung der Regelungsvorschläge, die sich unmittelbar auf die Herstellung von Interoperabilität durch den

³¹⁸ Vgl. BT-Dr. 12/4022, S. 13.

³¹⁹ *Marly*, Softwareüberlassungsverträge, Rdnr. 1173 unter Hinweis auf *OLG Düsseldorf*, CR 2001, 371 (372).

³²⁰ *Ensthaler*, Gewerblicher Rechtsschutz und Urheberrecht, S. 62.

³²¹ *Marly*, Softwareüberlassungsverträge, Rdnr. 1174.

³²² *Marly*, a.a.O.

³²³ *Kraßer*, Patentrecht, S. 27.

³²⁴ <http://de.internet.com/dynamic/print.php?id=2031233>.

Entwurf der Richtlinie des Europäischen Parlaments und des Rates über die Patentierbarkeit computerimplementierter Erfindungen beziehen.

Dokument	Regelungsvorschlag	Begründung
Richtlinienentwurf der EU-Kommission vom 20. Februar 2002 KOM(2002) 92 endg.	Art. 6 Konkurrenz zur Richtlinie 91/250/EWG „Zulässige Handlungen im Sinne der Richtlinie 91/250/EWG über den Rechtsschutz von Computerprogrammen durch das Urheberrecht, insbesondere der Vorschriften über die Dekompilierung und die Interoperabilität, oder im Sinne der Vorschriften über Marken oder Halbleitertopografien bleiben vom Patentschutz für Erfindungen aufgrund dieser Richtlinie unberührt.“	Artikel 6 stellt die Gültigkeit der in der Richtlinie 91/250/EWG enthaltenen Vorschriften für die Dekompilierung und Herstellung von Interoperabilität fest (Erwägungsgrund 18). Nicht berührt werden die Wettbewerbsvorschriften, insb. Artt. 81, 82 EGV (Erwägungsgrund 17). Die Patentanmeldung erfordere eine deutliche und vollständige Offenbarung der Erfindung. Im Fall missbräuchlicher Verwendung von Patenten könne auf Zwangslizenzen und die Regelungen des Wettbewerbsrechts zurückgegriffen werden (Begründung des RL-Entwurfs, S. 10).
Änderungsanträge des Europäischen Parlaments vom 24. September 2003 – P5_TA-Prov(2003)0402	Änderungsantrag zu Art. 6 „Rechte, die aus Patenten erwachsen, die für Erfindungen im Anwendungsbereich dieser Richtlinie erteilt werden, bleiben unberührt von urheberrechtlich zulässigen Handlungen gemäß Art. 5 und 6 der Richtlinie 91/250/EWG über den Rechtsschutz von Computerprogrammen, insbesondere gemäß den Vorschriften in Bezug auf die Dekompilierung und die Interoperabilität.“	„Unbeschränkter Patentschutz für Software könnte dazu führen, dass es gemäß dem Patentrecht unrechtmäßig ist, „reverse engineering“-Methoden anzuwenden, die von Softwareentwicklern eingesetzt werden, um Interoperabilität zu erzielen, wie dies derzeit im Rahmen der Ausnahmen der Richtlinie über Software-Urheberrecht gestattet ist. Daher müssen künftige EU-Rechtsvorschriften über Software-Patente eine ausdrückliche Ausnahme für Patentrechte beinhalten, um zu gewährleisten, dass die Entwickler von Software im Rahmen des Patentrechts weiterhin die gleichen Handlungen zur Verwirklichung der Interoperabilität durchführen können, wie dies heute im Rahmen des Urheberrechts gestattet ist“. (PE 327.249 – RR\501665DE.doc, S.16f.)
	Art. 6a (neu) ³²⁵ „Die Mitgliedsstaaten stellen sicher, dass in allen Fällen, in denen der Einsatz einer patentierten Technik für einen bedeutsamen Zweck wie die Konvertierung der in zwei verschie-	„Die Möglichkeit, Geräte miteinander zu verbinden, um sie interoperabel zu machen, ist eine Methode, um für offenen Netze zu sorgen und den Missbrauch marktbeherrschender Stellungen zu vermeiden. ... Das Patentrecht sollte nicht die Möglichkeit schaffen, diesen

³²⁵ Der Ausschuss des EP für Kultur, Jugend, Bildung, Medien und Sport machte einen Alternativvorschlag, der sich in der Abstimmung nicht durchsetzen konnte, aber mit Blick auf die Sicherstellung von Interoperabilität als tragfähiger Formulierungsvorschlag angesehen werden kann: „Rechte, die aus Patenten erwachsen, die für Erfindungen im Anwendungsbereich dieser Richtlinie erteilt werden, können die im Sinne der Richtlinie 91/250 über den Rechtsschutz von Computerprogrammen durch das Urheberrecht als Ausnahme gestatteten Handlungen, insbesondere die in Artikel 5 Absatz 2, Artikel 5 Absatz 3 dieser Richtlinie 91/250 beschriebenen Handlungen nicht berühren.“ Zur Begründung führte der Ausschuss an, dass das durch die Richtlinie 91/250 hergestellte sensible Gleichgewicht zwischen den Interessen des Inhabers des Urheberrechts und das Interesse der Herstellung von Interoperabilität, d.h. der Entwicklung interoperabler Programme, nicht gefährdet werden dürfe (PE 327.249 – RR\501665DE.doc, S.50.).

	denen Computersystemen oder – netzen verwendeten Konventionen benötigt wird, um die Kommunikation und den Austausch von Dateninhalten zwischen ihnen zu ermöglichen, diese Verwendung nicht als Patentverletzung gilt.“	Grundsatz auf Kosten des freien Wettbewerbs und der Nutzer umzustoßen.“ (PE 327.249 – RR\501665DE.doc, S.17.)
Rat der Europäischen Union – gemeinsamer Standpunkt vom 24. Mai 2004 - Ratsdokument 9713/04	Art. 6 Konkurrenz zur Richtlinie 91/250/EWG „ Rechte, die aus Patenten erwachsen, die für Erfindungen im Anwendungsbereich dieser Richtlinie erteilt werden, berühren nicht die urheberrechtlich zulässigen Handlungen gemäß Art. 5 und 6 der Richtlinie 91/250/EWG über den Rechtsschutz von Computerprogrammen, insbesondere gemäß den Vorschriften in Bezug auf die Dekompilierung und die Interoperabilität. “	„Rechte die aus Patenten erwachsen, die für Erfindungen im Anwendungsbereich dieser Richtlinie erteilt werden, bleiben unberührt von urheberrechtlich zulässigen Handlungen gemäß Artikel 5 und 6 der Richtlinie 91/250/EWG ..., insbesondere gemäß den Vorschriften in Bezug auf die Dekompilierung und die Interoperabilität. Insbesondere erfordern Handlungen, die gemäß Artikel 5 und 6 jener Richtlinie keine Genehmigung des Rechtsinhabers in Bezug auf dessen Urheberrechte an dem oder in Zusammenhang mit dem Computerprogramm erfordern, für die aber ohne Artikel 5 oder 6 jener Richtlinie eine solche Genehmigung erforderlich wäre, keine Genehmigung des Rechtsinhabers in Bezug auf die Patentrechte des Rechtsinhabers an dem oder im Zusammenhang mit dem Computerprogramm.“ (Ratsdokument 9713/04, S. 7).

Zwei unterschiedliche Regelungsmodelle stehen sich gegenüber. Während der Kommissionsentwurf nur eine Regelung zur Abgrenzung von Urheber- und Patentrecht vorsieht, schlägt das Europäische Parlament eine ergänzende Regelung in einem Art. 6a vor, die eine Ausnahme vom patentrechtlichen Ausschließlichkeitsrecht für die Fälle der Herstellung von Interoperabilität (Konvertierung und Kommunikation bzw. Austausch von Dateninhalten zwischen verschiedenen Computersystemen) vorsieht. Im Rahmen der Ratsverhandlungen im Vorfeld der Verabschiedung des gemeinsamen Standpunkts vom 24. Mai 2004, haben Luxemburg und Dänemark den regelungssystematischen Ansatz des Europäischen Parlaments aufgegriffen. Während Luxemburg dem Vorschlag des Europäischen Parlaments folgte und einen Ausnahmetatbestand in dem vorgenannten Sinne einbrachte, wählte Dänemark einen Regelungsansatz, der eher auf die Einführung einer Art Zwangslizenzsystem hindeutet, in dem die Lizenz sozusagen stillschweigend bzw. im Wege einer gesetzlichen Fiktion erteilt wird³²⁷.

Im Rahmen der weiteren Diskussion muss die Gewährleistung der zulässigen Durchführung von Maßnahmen zur Herstellung von Interoperabilität auch unter dem Patentrechtssystem berücksichtigt werden. Die Untersuchung hat insoweit gezeigt, dass die Unterschiedlichkeit der Schutzrechtssysteme Patent- und Urheberrecht die analoge Anwendung des urheberrechtlichen Dekompilierungsrechts auf das Patentrecht nicht zulässt, weshalb die im Richtlinienentwurf vorgesehene Regelung in Art. 6, wie nachfolgend ausgeführt, nicht ausreicht.³²⁸

³²⁷ „Member States shall ensure that wherever the use of a patented technique is needed for the sole purpose of ensuring interoperability such use is allowed under reasonable and non-discriminatory conditions.“

³²⁸ Siehe Kap. 9 .

10.4 Sicherstellung von Interoperabilität im Richtlinienentwurf des Kommission

Nachdem die Kommission den Richtlinienentwurf vorgelegt hatte, entwickelte sich eine rege Diskussion, in deren Verlauf das Europäische Parlament eine Reihe weit reichender Änderungsvorschläge in das Verfahren einbrachte. Die unterschiedlichen Regelungsansätze werden nachfolgend mit Blick auf deren Tauglichkeit zur Herstellung von Interoperabilität einer näheren Betrachtung unterzogen.

10.4.1 Kommissionsvorschlag

Art. 6 des Richtlinienentwurfs in der Fassung vom 20. Februar 2002 beschränkt sich in der Regelungsfunktion auf die Abgrenzung zu der Richtlinie 91/250/EWG, die urheberrechtliche Ausnahmetatbestände mit Blick auf die Durchführung von Dekompilierungsmaßnahmen zur Herstellung von Interoperabilität enthält. In der Begründung zur Richtlinie stellt die EU-Kommission kurz und knapp fest, dass Art. 6 die weitere Gültigkeit der Vorschriften der Richtlinie 91/250/EWG für die Dekompilierung und die Interoperabilität ausdrücklich bestätigt³²⁹. Es wird weiter ausgeführt:

„Daher sollte die Ausübung eines Patents an einer computerimplementierten Erfindung aufgrund der unterschiedlichen Schutzgegenstände beim Patent- und beim Urheberrecht und der Art der zulässigen Ausnahmen nicht mit den Freiheiten kollidieren, die das Urheberrecht den Softwareentwicklern nach der Richtlinie 91/250/EWG einräumt. Was die Entwicklung interoperierender Programme anbelangt, sollte darüber hinaus die Auflage, dass jedes Patent eine deutliche und vollständige Offenbarung der Erfindung enthalten muss, Fachleuten die Aufgabe erleichtern, ein Programm an ein anders bereits bestehendes Programm anzupassen, das patentierte Merkmale aufweist.“³³⁰

Im Umkehrschluss lässt sich der Formulierung entnehmen, dass der Richtlinienentwurf keine eigene Regelung patentrechtlicher Natur enthält, deren Zweck darauf gerichtet ist, Maßnahmen die der Dekompilierung ähnlich sind, d.h. die Anwendung von Methoden des Reverse Engineering, als zulässige Benutzung des Patents zu qualifizieren. Es muss daher bei der Feststellung verbleiben, dass nach der Fassung des Art. 6 Richtlinienentwurf vom 20. Februar 2002 das in der Richtlinie 91/250/EWG enthaltene Dekompilierungsrecht mit dem Ausschließlichkeitsrecht des Patentrechts kollidiert³³¹.

Der Hinweis auf die vollständige Offenbarung der Erfindung im Rahmen der Patentanmeldung verdeutlicht die Einschätzung der Kommission, dass ihrer Auffassung nach die Durchführung von Maßnahmen des Reverse Engineering im Anwendungsbereich des Richtlinienentwurfs grundsätzlich überflüssig ist. Wird die Erfindung vollständig offenbart, liegen alle zur Herstellung von Interoperabilität notwendigen Informationen vor, weshalb es an einem berechtigten Interesse Dritter an der Durchführung eines Reverse Engineering denknotwendig fehlt. Würde es in der Anmeldepraxis tatsächlich zu einer Offenbarung kommen, welche diejenigen Informationen enthält, die für die Entwicklung zur computerimplementierten Erfindung kommunikationsfähiger Anwendung erforderlich sind, könnte man sich den Folgerungen der Kommission anschließen. In der Praxis wird dieser Option aber nicht in nennenswertem Umfang genutzt³³². Zwar sehen

³²⁹ Vorschlag für eine Richtlinie des Europäischen Parlaments und des Rates über die Patentierbarkeit computerimplementierter Erfindungen, KOM (2002) 92 endg., S. 17.

³³⁰ Vorschlag für eine Richtlinie des Europäischen Parlaments und des Rates über die Patentierbarkeit computerimplementierter Erfindungen, KOM (2002) 92 endg., S. 9f ; kritisch hierzu *Cornish/Llewelyn*, Intellectual Property, 19 – 33.

³³¹ Siehe 8.3.5.2 .

³³² Vg. 9.1 -

die Prüfrichtlinien von EPA und DPMA die Einreichung des Quellcode im Rahmen der Offenbarung der Erfindung vor, in der Praxis machen die Anmelder von dieser Möglichkeit, wie in diesem Gutachten aufgezeigt, jedoch ganz überwiegend keinen Gebrauch. Die Hinweis in der Begründung zum Richtlinienentwurf läuft somit ins Leere. Will man die Herstellung von Interoperabilität auch weiterhin gewährleisten, muss der Gleichlauf zwischen Urheber- und Patentrecht hergestellt werden, wenn die betreffenden Schutzgegenstände sowohl urheber- als auch patentrechtlichen Schutz zu erlangen vermögen. Werden die entsprechenden Informationen im Patentanmeldeverfahren tatsächlich überwiegend nicht offen gelegt, so greift die in Art. 6 Richtlinienentwurf zu kurz.

Kommt es zu missbräuchlicher Ausübung von Patentrechten, so ermöglicht nach Auffassung der Kommission die Erteilung von Zwangslizenzen, bzw. das Ergreifen kartellrechtlicher Korrekturmaßnahmen die Gewährleistung des urheberrechtlichen Dekompilierrechts³³³. Auch in diesem Punkt weichen Rechtslage und Rechtspraxis nicht unerheblich voneinander ab. Im Rahmen dieses Gutachtens konnte nicht valide festgestellt werden, inwieweit die Eignung von Zwangslizenzen und kartellrechtlicher Instrumenten tatsächlich gegeben ist, um den Anforderungen der Softwarebranche Genüge tun zu können. Es hat sich jedoch gezeigt, dass beide Instrumente zu schwerfällig sind, als dass sie zeitnah zu rechtssicheren und den Rahmenbedingungen der Softwarebranche gerecht werdenden Ergebnissen führen könnten.

Art. 6 Richtlinienentwurf i.d.F. vom 20. Februar 2002 ist daher nicht geeignet, Interoperabilität sicherzustellen, da das Patentrecht in der Lage ist, urheberrechtlich bestehende Dekompilierbefugnisse zu blockieren.

10.4.2 Änderungsvorschläge des Europäischen Parlamentes

Aus den Änderungsanträgen des Europäischen Parlamentes geht die besondere Bedeutung der Herstellung von Interoperabilität, auch unter dem Schutzrechtssystem des Patentrechts, hervor. Die Änderungsanträge 13, 19 und 23 sind darauf gerichtet die beschriebenen Unstimmigkeiten zu beseitigen³³⁴. In den Begründungen zu den Änderungsanträgen wird die berechtigte Besorgnis zum Ausdruck gebracht, unbeschränkter Patentschutz für computerimplementierte Erfindungen könnte zur Unrechtmäßigkeit der Anwendung von „Reverse Engineering“-Methoden führen. Vorgeschlagen wurde daher in einem neu einzufügenden Art. 6a die ausdrückliche Aufnahme entsprechender Formulierungen, die für die Fälle, in denen Interoperabilität zwischen unterschiedlichen Computersystemen hergestellt werden soll, Ausnahmen vom Patentschutz enthalten. Durch Konvertierung soll der Austausch von Dateninhalten sowie die Kommunikation zwischen Computersystemen ermöglicht werden. Zur Begründung führt das Europäische Parlament an, dass gerade offene Netze darauf angewiesen sind, dass Geräte interoperabel miteinander verbunden werden können und auf diesem Wege dem Missbrauch marktbeherrschender Stellungen entgegengewirkt werden kann. Regelungssystematisch zutreffend soll die Sicherstellung der in Art. 6a genannten Voraussetzungen den Mitgliedsstaaten überlassen werden, da das EPÜ selbst keine Regelungen zum Benutzungsrecht und zu Verletzung von Patentrechten enthält, sondern diesbezüglich auf das nach IPR-Grundsätzen berufene Recht eines Vertragsstaates verweist.

Erforderlich ist diese Vorgehensweise schon aus dem Grund der Nichtherstellbarkeit von Analogien über die Schutzrechtssysteme hinaus. Eine entsprechende Anwendung des urheberrechtlichen Dekompilierrechts auf patentrechtliche Sachverhalte scheidet, wie nachgewiesen aus, so dass die Übertragung dieses Grundsatzes nach dem Grundsatz *mutandis mutatis*

³³³ Vorschlag für eine Richtlinie des Europäischen Parlaments und des Rates über die Patentierbarkeit computerimplementierter Erfindungen, KOM (2002) 92 endg., S. 10.

³³⁴ Bericht über den Vorschlag für eine Richtlinie des Europäischen Parlamentes und des Rates über die Patentierbarkeit computerimplementierter Erfindungen (KOM(2002) 92 – C5-0082/2002 – 2002/0047(COD)) .- A5-0238/2003 endg.; S. 10, 12, 16.

nicht möglich ist³³⁵. Insoweit kann auf die zutreffende Begründung zu Änderungsantrag 13 verwiesen werden³³⁶.

In diesem Sinne weitergehend, ist die im Ausschuss für Industrie, Außenhandel, Forschung und Energie entwickelte Begründung zu Artt. 5 lit. c³³⁷ und d³³⁸ Richtlinienentwurf. Unter Hinweis auf die Monopolwirkung des Patents, das im Rahmen gewerblicher Anwendung dem Dekompilierungsrecht des Urheberrechts diametral gegenübersteht, so argumentiert der Ausschuss, dürfe das Innovationsstreben von Konkurrenten des Patentinhabers nicht unterbunden werden³³⁹. Zugleich müsse durch einen neu einzufügenden Art. 6a Richtlinienentwurf³⁴⁰ die Möglichkeit offen gehalten werden, Geräte miteinander verbinden zu können, um sie auf diesem Wege interoperabel zu machen³⁴¹. Aus der Stellungnahme des Ausschusses für Kultur, Jugend, Bildung, Medien und Sport geht somit hervor, dass man Parallelität zwischen Urheber- und Patentrecht herstellen möchte³⁴². Zutreffend wurde erkannt, dass nur klare Regelungen zu mehr Rechtssicherheit führen können. In dem Maße in dem neue Auslegungsspielräume eröffnet werden, werden von Patentinhabern Versuche unternommen werden, sich dieser Verpflichtungen zu entziehen.

Ob durch die Änderungsvorschläge Fallgestaltungen, die die Herstellung von Interoperabilität von Softwarelösungen betreffen hinreichend erfasst werden, die auf die Kommunikationsfähigkeit mit sog. „Embedded Systems“ angewiesen sind ist dagegen fraglich, da wie aufgezeigt, Maßnahmen des Reverse Engineering in diesen Konstellationen weitgehend technisch nicht zu den gewünschten Ergebnissen führen³⁴³. Allein durch die Offenlegung der Schnittstelleninformationen in der Patentanmeldung kann bei sog. „Embedded Systems“ Interoperabilität gewährleistet werden.

Somit liegt der Schluss nahe, dass insgesamt Interoperabilität durch die Veröffentlichung von Schnittstelleninformationen gewährleistet werden kann. Aus technischer Sicht wird man jedoch im Regelfall ein Reverse Engineering vorziehen. Es steht zu erwarten, dass die prüfenden Patentämter personell und fachlich mit der Feststellung der Eignung der hinterlegten Informationen

³³⁵ Den speziell urheberrechtlichen Charakter des in der EU-Richtlinie über den Rechtsschutz von Computerprogrammen vom 14. Mai 1991, ABI EG Nr. L 122/42 hervorhebend, *Leith*, JILT 2000, p.12 - http://www2.warwick.ac.uk/fac/soc/law/elj/jilt/2000_2/leith1/ (letzter Abruf 5. Oktober 2004)

³³⁶ Bericht über den Vorschlag für eine Richtlinie des Europäischen Parlamentes und des Rates über die Patentierbarkeit computerimplementierter Erfindungen (KOM(2002) 92 – C5-0082/2002 – 2002/0047(COD)) .- A5-0238/2003 endg.; S. 12f.

³³⁷ (c) Die Mitgliedsstaaten stellen sicher, dass die Verwendung eines Computerprogramms für Zwecke, die nicht zum Gegenstand des Patents gehören, keine direkte oder indirekte Patentverletzung darstellen kann.

³³⁸ (d) Die Mitgliedsstaaten stellen sicher, dass in allen Fällen, in denen in einem Patentsanspruch 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.

³³⁹ Bericht über den Vorschlag für eine Richtlinie des Europäischen Parlamentes und des Rates über die Patentierbarkeit computerimplementierter Erfindungen (KOM(2002) 92 – C5-0082/2002 – 2002/0047(COD)) .- A5-0238/2003 endg.; S. 35.

³⁴⁰ Siehe Tabelle unter 10.4 .

³⁴¹ Bericht über den Vorschlag für eine Richtlinie des Europäischen Parlamentes und des Rates über die Patentierbarkeit computerimplementierter Erfindungen (KOM(2002) 92 – C5-0082/2002 – 2002/0047(COD)) .- A5-0238/2003 endg.; S. 36.

³⁴² Bericht über den Vorschlag für eine Richtlinie des Europäischen Parlamentes und des Rates über die Patentierbarkeit computerimplementierter Erfindungen (KOM(2002) 92 – C5-0082/2002 – 2002/0047(COD)) .- A5-0238/2003 endg.; S. 50.

³⁴³ Vgl. 8.4 .

zu Zwecken der Programmierung überfordert sein werden³⁴⁴. Unter Praktikabilitäts Gesichtspunkten ist daher im Ergebnis ein Nebeneinander von Reverse Engineering und Offenlegung von Schnittstelleninformationen notwendig, um so ausreichende Flexibilität der gesetzlichen Regelungen herzustellen, damit alle Fallgestaltungen der Herstellung von Interoperabilität erfasst werden können.

10.4.3 Kompromissvorschläge

Die unter 10.4 in der Tabelle wiedergegebenen Kompromissvorschläge der Mitgliedsstaaten Luxemburg und Dänemark gehen nicht über die Vorschläge des Europäischen Parlament hinaus. Luxemburg greift den Vorschlag des Europäischen Parlaments wortgleich auf³⁴⁵. Aus diesem Grund kann auf die oben wiedergegebene Begründung verwiesen werden.

Einen neuen Weg schlägt der von Dänemark eingebrachte Vorschlag ein. Muss ein Patent zur Herstellung von Interoperabilität benutzt werden, ist die Benutzung unter vernünftigen und nicht diskriminierenden Bedingungen zu gestatten. Im Kern wird die Einführung einer Art Zwangslizenzsystem vorgeschlagen, da die Lizenz unter den genannten Voraussetzungen zu erteilen ist, bzw. stillschweigend im Wege einer gesetzlichen Fiktion als erteilt vorausgesetzt wird³⁴⁶. Eine weitergehende Analyse lässt der doch sehr vage formulierte dänische Kompromissvorschlag nicht zu, zumal in dem betreffenden Ratsdokument keine weitere Informationen hinterlegt wurden, die Rückschlüsse auf die Beweggründe zuließen, die Dänemark dem Formulierungsvorschlag zugrunde gelegt hat.

Entscheidender Nachteil dieses Regelungsansatzes ist, dass die bestehenden Rechtsunsicherheit nicht beseitigt werden. Wenn man von einer Anlehnung an das System der Zwangslizenz ausgeht, wird der Dritte zuvor beim Patentinhaber um die Erteilung einer Lizenz nachsuchen müssen. Auch wenn dem Dritten dies zumutbar ist und er im Zweifel die Notwendigkeit der Benutzung des Patents zur Herstellung von Interoperabilität nachweisen können, so wird jedoch letztlich der Streit um die Frage entfacht werden, ob die vom Patentinhaber genannten Lizenzbedingungen nicht diskriminierend sind. Solange in diesem Punkt keine Beweislastumkehr zu Lasten des Patentinhabers vorgesehen wird, und dies ist dem Vorschlag in der derzeitigen Fassung nicht zu entnehmen, wird diese Hürde de facto dazu führen, dass Dritte entsprechende Lizenzrechte im Zweifel wegen des Prozess- und Kostenrisikos sowie der Zeitkomponente nicht gerichtlich durchsetzen werden.

Diese Kompromissvorschläge sind nicht in der Entscheidung des Rates der Europäischen Union aufgegriffen worden.

10.4.4 Gemeinsamer Standpunkt

In dem gemeinsamen Standpunkt vom 24. Mai 2004 enthaltene Formulierungsvorschlag für die die Interoperabilität betreffenden Regelung des Richtlinienvorschlages³⁴⁷ fällt weit hinter die Vorschläge von Europäischem Parlament und den Kompromissvorschlägen von Luxemburg

³⁴⁴ Auf diese Problematik unter dem Blickwinkel der Pflicht zur Überlassung von Quellcode im Rahmen der Softwarehinterlegung hinweisen, *Hoeren*, CR 2004, 721ff.

³⁴⁵ Siehe englische Fassung des Bericht über den Vorschlag für eine Richtlinie des Europäischen Parlamentes und des Rates über die Patentierbarkeit computerimplementierter Erfindungen (KOM(2002) 92 – C5-0082/2002 – 2002/0047(COD)) .- A5-0238/2003 endg.; S. 18 und Ratsdokument 9051/04, S. 10 FN. 10.

³⁴⁶ „Member States shall ensure that wherever the use of a patented technique is needed for the sole purpose of ensuring interoperability such use is allowed under reasonable and non-discriminatory conditions.”

³⁴⁷ Ratsdokument 9713/04.

und Dänemark zurück, indem die ursprüngliche Formulierung des Art. 6 konkreter ausgestaltet, aber wiederum auf die reine Abgrenzung zur Richtlinie 91/250/EWG beschränkt wurde und damit die Einfügung eines die patentrechtlichen Benutzungsrechte ausdrücklich regelnden Art. 6a Richtlinienentwurf abgelehnt wird. Zweck der Regelung soll nämlich primär sein, dass derjenige, dem ein urheberrechtliches Dekompilierungsrecht zusteht, nicht erneut beim Patentinhaber um eine Genehmigung der Durchführung der betreffenden Maßnahmen nachsuchen muss³⁴⁸.

11 Zusammenfassung und Empfehlungen

Bei Hochtechnologiemärkten besteht die latente Gefahr einer faktischen Zutrittskontrolle und –begrenzung durch Patente und die Lizenzvergabepolitik von Patentinhabern von Schlüsseltechnologien. Aus diesem Grund müssen die Schutzrechtssysteme Mechanismen bereithalten, um Monopolbildungen auf patentfremden Gebieten verhindern zu können.

Im Bereich der Patentierung computerimplementierter Erfindungen konnte in den letzten Jahren eine deutliche Ausweitung des Patenschutzes beobachtet werden. Durch die wachsende Zahl von Patenten für computerimplementierte Erfindungen – schätzungsweise hat alleine das EPA 250.000 solcher Patente erteilt³⁴⁹ - wird der Druck auf die Softwarebranche deutlich erhöht, da im Rahmen der Softwareentwicklung eine Vielzahl von Rechten Dritter zu berücksichtigen ist, damit Patentverletzungen vermieden werden.

Die Analyse der Patenterteilungspraxis von DPMA und EPA hat gezeigt, dass sich hierunter in besonderen Maße vermehrt Verfahrenspatente befinden, was zur Folge hat, dass DV-Programme unter dem Dach einer computerimplementierten Erfindung am Patentschutz des Verfahrens teilhaben.

Auf internationaler Ebene wird der Anpassungsdruck auf das deutsche und europäische Patentrechtssystem stetig erhöht, da das EPA insbesondere die U.S.-amerikanische Patentierungspraxis als Leitsystem für eigene Entscheidungen heranzieht und die Auslegung von Vorschriften des EPÜ entsprechend anpasst. Der Richtlinien-Entwurf in der Fassung vom 24. Mai 2004 tritt dieser Entwicklung in erfreulicher Weise entgegen.

Durch diese Praxis wird das an sich nach EPÜ und deutschem PatG bestehende Verbot der Patentierung von Software „als solcher“ nahezu ausgehöhlt.

Soll das Patentrecht für die deutsche Softwareindustrie, die sich im wesentlichen auf die Entwicklung von Individualsoftware spezialisiert hat, die Möglichkeit offen halten, Entwicklungen unter Ausschluss von Patentverletzungen zu ermöglichen, müssen Instrumente zur Herstellung von Interoperabilität bereitgestellt werden.

Interoperabilität verlangt insbesondere die Offenlegung von Informationen über Software- und Kommunikationsschnittstellen. In § 34 PatG ist ein Werkzeug enthalten, das entsprechend praxistaugliche Ergebnisse zu liefern vermag. Der Anmelder einer Erfindung erhält die Möglichkeit, die erforderliche Offenlegung der Erfindung u.a. durch die Mitteilung des Quellcodes zu bewirken. Allerdings wird diese Option in der Praxis bislang kaum genutzt und birgt die Gefahr in sich, dass für eine Programmierung nicht taugliche Beschreibungen hinterlegt werden.

³⁴⁸ Ratsdokument 9713/04, S. 7.

³⁴⁹ *Hilty*, MMR 2003, 3 (6) FN. 36.

Notwendig wird die Herstellung von Interoperabilität, wenn Softwareschnittstellen, Konvertierungsprogramme und Nachrichten-Broker die Kommunikation zwischen verschiedenen Anwendungen sowie Betriebssystemen und Applikationssoftware herstellen sollen.

Da die Umsetzung der Softwareschnittstellen im Regelfall unselbständige Teile eines DV-Programms sind, welches wiederum Bestandteil einer computerimplementierten Erfindung ist, stellt sich die Frage, ob der Entwickler, für den Fall, dass der Patentinhaber weder SDK's noch sonstige Schnittstelleninformationen bereitstellt, berechtigt ist, selbsttätig die entsprechenden Informationen auf dem Wege des Reverse Engineering zu erlangen. Gleiches gilt für Konvertierungsprogramme und Nachrichten-Broker, da auch hier die Kommunikation zwischen den zu verbindenden Systemen über Softwarelösungen realisiert werden.

Aus Sicht des Patentrechts sind dem Reverse Engineering enge Grenzen gesetzt. Da im Bereich der Softwareentwicklung diese Methode zur Informationsgewinnung im Regelfall gewerblicher Natur sein wird, blockiert das Patentrecht diese Option zur Herstellung von Interoperabilität, da die entsprechenden Handlungen als Patentverletzung zu qualifizieren sind.

Ob die Lizenzerteilungspolitik von marktstarken Unternehmen, wie das Beispiel des CIFS-Datenformats gezeigt hat, als adäquates Substitut patentrechtlicher Befugnisse zur Durchführung von Reverse Engineering, betrachtet werden können, ist äußerst fraglich. So dürfte der Zwang zur Offenlegung von Kundenstämmen eine deutliche Barriere errichten, die eine Vielzahl von Unternehmen vom Abschluss entsprechender Lizenzvereinbarungen abhalten wird.

In naher Zukunft wird Embedded Systems und Trusted Computing-Technologien, aufgrund der rasant fortschreitenden Computerisierung von Anwendungen in Produkten und einem wachsenden Sicherheitsbedürfnis, erheblichen Bedeutung zukommen. Aus technischen Gründen wird die Herstellung von Interoperabilität durch Reverse Engineering bei Embedded Systems ausscheiden. Daher kann in diesem Bereich der für die Entwickler von Applikationssoftware notwendige Informationsfluss nur und ausschließlich über die Offenlegung der Schnittstelleninformationen erfolgen können.

Anders ist die Situation im Bereich des Trusted Computing zu bewerten. In diesem Fall spielt die Frage der Standardisierung/Normung und hier die Konsensfindung im Markt eine bedeutende Rolle. Dies führt dazu, dass geradezu zwangsläufig die für die Sicherstellung von Interoperabilität notwendigen Informationen zur Verfügung gestellt werden.

Der Richtlinien-Entwurf über den Patentschutz für computerimplementierte Erfindungen vermeint die Interoperabilität dadurch sichergestellt zu haben, dass einerseits das urheberrechtlich verankert Dekompilierungsrecht unberührt bleibt (Art. 6 des Entwurfs). Wegen der z.T. gravierenden systematischen Unterschiede, die zwischen Urheber- und Patentrecht bestehen, scheidet nach hier vertretener Auffassung eine „entsprechende“ Anwendung des urheberrechtlichen Dekompilierungsrechts auf Fallgestaltungen des Patentrechts aus.

Schließlich stützen die aus dem rechtsvergleichenden Überlegungen gewonnenen Erkenntnisse die Folgerung, dass nicht auf das Dekompilierungsrecht des Urheberrechts rekurriert werden kann, da in der Vergangenheit der Rechtsschutz von Computerprogrammen nahezu ausschließlich mittels des Urheberrechts gewährleistet wurde. Erst aufgrund des Bedeutungszuwachses, den das Patentrecht in dieser Hinsicht erfahren hat, fehlt es an entsprechenden Erfahrungswerten und durch die Rechtsprechung gesetzter Eckwerte. Die hierdurch entstehende Rechtsunsicherheit wirkt, angesichts der Bedeutung der Softwarebranche für die deutsche und europäische Wirtschaft, in besonderer Weise kontraproduktiv und auf die Softwarebranche destabilisierend.

Ferner wurde nachgewiesen, dass ein nicht lizenzierungsbereiter Patentinhaber weder durch die Androhung der Erteilung von Zwangslizenzen, noch durch kartellrechtliche Maßnahmen zeitnah dazu bewegt werden kann, einen die Interoperabilität sicherstellenden Zustand herbeizuführen. Beide Verfahren sind nicht in der Lage kurzfristig Abhilfe zu schaffen, so dass sie sich, angesichts der kurzen Entwicklungs- und Produktzyklen in der Softwarebranche, als unwirksam erweisen werden. Bis eine Entscheidung getroffen wird, dürfte im Regelfall entweder der entsprechende Markt bereits besetzt sein, oder der Entwickler hat die entsprechenden Bemühungen ohnehin eingestellt.

Soll das Ziel der Sicherstellung von Interoperabilität erreicht werden, welches für die Erhaltung der Vitalität des deutschen und europäischen Softwaremarktes von entscheidender Bedeutung ist, muss der Richtlinienentwurf zumindest die vorgehend angesprochenen Änderungsvorschläge des Europäischen Parlamentes aufgreifen. Nicht ausreichend ist dies für den Bereich von Embedded Systems, so dass alternativ die Überlegung angestellt werden kann, in einem ersten Schritt § 11 PatG dahingehend zu ändern, dass Handlungen des Reverse Engineering zu Zwecken der Herstellung von Interoperabilität das bestehende Ausschließungsrecht des Patentinhabers insoweit einschränken. Zugleich bietet sich an, eine entsprechende Regelung in den Richtlinienentwurf aufzunehmen. Die für den Patentanmelder im Rahmen des Anmeldeverfahrens bestehende Option der Offenlegung der Erfindung durch Vorlage von Quellcodes nachzukommen ist zu vage und stellt nicht mit der notwendigen Sicherheit die Interoperabilität sicher, zumal die Patentämter weder personell noch sachlich über die Mittel verfügen, dass hinterlegt Material auf seine Güte und Tauglichkeit im Rahmen einer möglichen künftigen Programmierung interoperabler Software zu überprüfen.

Soll Interoperabilität mit hinreichender Rechtssicherheit hergestellt werden, muss die Regelung klar und eindeutig gefasst werden, um möglichen Auslegungsschwierigkeiten vorzubeugen. Die Untersuchung hat gezeigt, dass der Richtlinientext insoweit zweigleisig angelegt werden muss um klarzustellen, dass sowohl das Urheber- wie auch das Patentrecht die Durchführung von Maßnahmen zur Herstellung von Interoperabilität gestattet, ohne das zuvor eine entsprechende spezielle Lizenz vom Rechteinhaber erteilt werden muss. Diesen Anforderungen genügt die vom Europäischen Parlament vorgeschlagene Neufassung des Art. 6 sowie des neu einzufügenden Art. 6a. Alle anderen untersuchten Regelungsvorschläge führen nicht zu den erwünschten Ergebnissen.

Regelungsvorschlag:

Änderungsantrag zu Art. 6

„Rechte, die aus Patenten erwachsen, die für Erfindungen im Anwendungsbereich dieser Richtlinie erteilt werden, bleiben unberührt von urheberrechtlich zulässigen Handlungen gemäß Art. 5 und 6 der Richtlinie 91/250/EWG über den Rechtsschutz von Computerprogrammen, insbesondere gemäß den Vorschriften in Bezug auf die Dekompilierung und die Interoperabilität.“

Änderungsantrag Einfügung Art. 6a

„Die Mitgliedsstaaten stellen sicher, dass in allen Fällen, in denen der Einsatz einer patentierten Technik nur zum Zweck der Konvertierung der in zwei verschiedenen Computersystemen oder –netzen verwendeten Konventionen benötigt wird, um die Kommunikation und den Austausch von Dateninhalten zwischen ihnen zu ermöglichen, diese Verwendung nicht als Patentverletzung gilt.“

Als weitere Option zur Offenlegung von Schnittstelleninformationen zur Herstellung von Interoperabilität wurde eine entsprechende Verpflichtung im Anmeldeverfahren diskutiert. Im Gegensatz zu der vorgehend ausgesprochenen Regelungsempfehlung, birgt diese Variante eine Reihe von Gefahren, die in der Praxis kaum wirksam kontrolliert werden können. Angesichts sinkender personeller und sachlicher Ausstattung der die Patentanmeldung prüfenden Patentämter, würde auf diese eine erhebliche Mehrbelastung zukommen, wenn diese die Verwendbarkeit der offen gelegten Schnittstelleninformationen für Zwecke der Programmierung interoperabler Software zu prüfen hätten. Erfolgt diese Prüfung nicht, wäre interoperabilitätsschädlichem Missbrauch Tor und Tür geöffnet und der die Information nachsuchende letztlich wieder auf das Reverse Engineering verwiesen. Einer solchen Regelung kann daher nur komplementäre bzw. appellative Wirkung zukommen. Aufgegriffen werden kann ein Formulierungsvorschlag des Ausschuss für Industrie, Außenhandel, Forschung und Energie des Europäischen Parlaments zur Ergänzung des Art. 5 Form des Patentanspruchs um eine Regelung zu obligatorisch offen zulegenden Informationen

Die Mitgliedsstaaten stellen sicher, dass in allen Fällen, in denen in einem Patentsanspruch 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.

In der Begründung zu dieser Regelung muss eine Klarstellung enthalten sein, die die Möglichkeit eröffnet, im Rahmen der Offenlegung der Dokumentation ersatzweise sog. SDK's hinterlegen zu können³⁵⁰, da Software in sehr unterschiedliche Umgebungssysteme eingebunden wird und genutzt werden kann, weshalb keine einheitlichen Methoden der Beschreibung von Softwareschnittstellen zur Anwendung kommen. Statt dessen kann eine gute Beschreibung der Schnittstelle einer Software durch ein sog. "Software Development Kit (SDK)" erfolgen. Ein Software Development Kit ist eine Sammlung von Programmen und Dokumentationen zu einer bestimmten Software, die es Software-Entwicklern erleichtern soll, eigene darauf basierende Anwendungen zu erstellen.

³⁵⁰ Siehe 4.3 .

Literaturverzeichnis

- Anders, Winfried** Wie viel technischen Charakter braucht eine computerimplementierte Geschäftsmethode, um auf erfinderischer Tätigkeit zu beruhen?, GRUR 2001, S. 555 ff
- ders.** Erfindungsgegenstand mit technischen und nichttechnischen Merkmalen, GRUR 2004, S. 461 ff.
- Beise, Marc** Die Welthandelsorganisation (WTO) – Funktion, Status, Organisation, Baden-Baden 2001
- Blumberg, Hartmut/ Pohlmann, Norbert** Der IT-Sicherheitsleitfaden - Das Pflichtenheft zur Implementierung von IT-Sicherheitsstandards im Unternehmen, Bonn 2004
- Brandl, Hans/ Rosteck, Thomas** Technik, Implementierung und Anwendung des Trusted Computing Group-Standards (TCG), DuD 2004, S. 529 ff.
- Busche, Jan** Patentschutz für computerimplementierte Erfindungen, 2003 (<http://www.jura.uni-duesseldorf.de/dozenten/busche/Vortragmat/softwarevortragEPA.pdf> (letzter Abruf 17. September 2004))
- Busse, Rudolf** Patentgesetz – Kommentar, 6. Aufl. Berlin 2003 (zit.: Bearbeiter in: Busse, Patentgesetz §. Rdnr. ..)
- Cohen, Julie/ Lemley, Mark** Patent Scope and Innovation in the Software Industry, 89 Calif.L.Rev. 1ff.
- Cornish; William/ Llewelyn, David** Intellectual Property: Patents, Copyright, Trade Marks and Allied Rights, London 2003
- Detlefs, Adrian** Recherche zur elektronischen Gesundheitskarte und zum digitalen Krankenhaus, Diplomarbeit Fachhochschule Gelsenkirchen, 2004
- Dohrmann, Thomas** Rechtsschutz für Computerprogramme und Software in Indien, Hannover 2001
- Dreher, Meinrad** Die Verweigerung des Zugangs zu einer wesentlichen Einrichtung als Missbrauch der Marktbeherrschung, DB 1999, S. 833 - 839
- Emmerich, Volker** Kartellrecht, 9. Aufl. München 2001
- ders.** Unlauterer Wettbewerb, 6. Aufl. München 2002
- Ensthaler, Jürgen** Gewerblicher Rechtsschutz und Urheberrecht, 2. Aufl. Berlin 2003
- Fischer, Daniel** Die Behandlung technischer Handelshemmnisse im Welthandelsrecht, Heidelberg 2004

- Fleischer, Holger/ Doege, Niels** Der Fall United States v. Microsoft . Zwischenbilanz eines kartellrechtlichen Jahrhundertverfahrens, WuW 2000, S. 705 – 717
- Fleischer, Holger/ Weyer, Hartmut** Neues zur „essential facilities“-Doktrin im Europäischen Wettbewerbsrecht – Eine Besprechung der Bronner-Entscheidung des EuGH, WuW 1999, S. 350 - 363
- Gassner, Ulrich** Grundzüge des Kartellrechts, München 1999
- Goebel, Frank Peter** Technizität – zum Patentschutz für programmbezogene Erfindungen nach der jüngeren deutschen Erteilungs- und Entscheidungspraxis, in: Festschrift für Rudolf Nirk zum 70. Geburtstag, Hrsg. v. Bruchhausen, Karl/ Hefermehl, Wolfgang/ Hommelhoff, Peter/ Messer, Herbert, München 1992, S. 357 ff.
- Haas, Franz-Werner** Das TRIPS-Abkommen: Geistiges Eigentum als Gegenstand des Welthandelsrechts – Das TRIPS-Abkommen und seine Auswirkungen unter Berücksichtigung der Rechtsordnung der Europäischen Gemeinschaft, Baden-Baden 2004
- Hilty, Reto** Der Softwarevertrag – ein Blick in die Zukunft – Konsequenzen der trägerlosen Nutzung und des patentrechtlichen Schutzes von Software, MMR 2003, S. 3 ff.
- Hoeren, Thomas** Die Pflicht zur Überlassung des Quellcodes – Eine liberale Lösung des BGH und ihre Folgen, CR 2004, S. 721 - 724
- Horns, Axel** Anmerkungen zu begrifflichen Fragen des Softwareschutzes, GRUR 2001, S. 1 ff.
- Intel Corporation** LaGrande: Technology Architectural Overview; Intel White Paper, September 2003
- Keukenschrijver, Alfred** Sind bei der Beurteilung der erfinderischen Tätigkeit sämtliche Merkmale im Patentanspruch gleichermaßen zu berücksichtigen?, in: Festschrift für Reimer König zum 70. Geburtstag Hrsg. v. Ann, Christoph/ Anders, Wilfried/ Dreiss, Uwe/ Jestaedt, Bernhard/ Stauder, Dieter, S. 255 ff.
- Kiewsewetter-Köbinger, Swen** Über die Patentprüfung von Programmen für Datenverarbeitungsanlagen, GRUR 2001, S. 185 ff.
- Kilian, Wolfgang/ Heussen, Benno** ComputerrechtsHandbuch, Loseblatt, Stand Oktober 2003 München (zit.: *Bearbeiter* in: Kilian/Heussen Computerrechts-Handbuch, Kap. ... Rdnr. ...)
- Koenig, Christian/ Kühling, Jürgen** Funktionsfähiger Wettbewerb und Regulierungsperspektiven auf den Telekommunikationsmärkten, WuW 2000, S. 596 - 610
- Kraßer, Rudolf** Patentrecht, 5. Aufl. 2004 München

- Kübel, Constanze** Zwangslizenzen im Immaterialgüter- und Wettbewerbsrecht – Eine Untersuchung zu Patenten und Urheberrechten bei technischen Normen, Köln 2004
- Leith, Philip** Software Utility Models and SMEs, *The Journal of Information, Law and Technology* 2000, (http://www2.warwick.ac.uk/fac/soc/law/elj/jilt/2000_2/leith1/ letzter Abruf 5. Oktober 2004)
- Marly, Jochen** Softwareüberlassungsverträge, 4. Aufl. München 2004
- Meier-Wahl, Marc/ Wrobel, Ralph Michael** Wettbewerbsregulierung in einem dynamischen Markt, *Der Fall Microsoft*, WuW 1999, S. 28 - 33
- Mellius, Klaus-J.** Zur Patentfähigkeit von Programmen für die Datenverarbeitung, GRUR 1998, S. 843 ff.
- Mestmäcker, Ernst-Joachim/ Schweitzer, Heike** Europäisches Wettbewerbsrecht, 2. Aufl. München 2004
- Miller, Arthur/ Davis, Michael** Intellectual Property – Patents, Trademarks and Copyright, 2nd. Ed., St. Paul, Minn. 1990
- Nack, Ralph/ Phélip, Bruno** Bericht über die Diplomatische Konferenz zur Revision des Europäischen Patentübereinkommens München 20- - 29. November 2000, GRUR Int. 2001, S. 322 ff
- Ohly, Ansgar** Software und Geschäftsmethoden im Patentrecht, CR 2001, S. 809 ff.
- Peter, Marcus** Russia and the WTO – Comparative analysis of Russian and WTO law, Baden-Baden 2004
- Petry, Nicolaus** Fuzzy Logik und neuronale Netze, JurPC Web-Dok. 187/1999
- Pierson, Matthias** Der Schutz der Programme für die Datenverarbeitung im System des Immaterialgüterschutzes, Pfaffenweiler 1991 Diss. Marburg 1991
- ders.** Softwarepatente – technische und patentrechtliche Grundlagen, JurPC Web-Dok. 18/2004
- ders.** Softwarepatente – Meilensteine und Kategorien im Spiegel der patentrechtlichen Rechtsprechung, JurPC Web-Dok. 163/2004
- Probster, W.E** Rechnernetze - Technik, Protokoll, Systeme und Anwendungen, München 2002
- Ravicher; Daniel** Patents – Why Free/Open Source Software might have less to fear than non-free software (http://groklaw.net/pdf/Patents.pdf - letzter Abruf 16. September 2004)
- Rehbinder, Manfred** Urheberrecht, 13. Aufl. München 2004

- Rübel, Clemens** Patentschutz bei Reparatur- und Ersatzteilmfällen, GRUR 2002, S. 561 - 565
- Sadeghi, Ahmad-Reza/ Stüble, Christian** Taming "Trusted Computing" by Operating System Design, Proceedings of the 4th International Workshop on Information Security Applications (WISA'03), Cheju Island, Korea, 2003
- Samuelson, Pamela/ Scotchmer, Suzanne** The Law & Economics of Reverse Engineering, 111 Yale.L.J. 1575.
- Schallbruch, Martin** Trusted Computing – Chance für eine sichere Informationsgesellschaft, DuD 2004, S. 519 – 520
- Scheffler, Dietrich** Die (ungenutzten) Möglichkeiten des Rechtsinstituts der Zwangslizenz, GRUR 2003, S. 97 – 105
- Schramm, Carl (Hrsg.)** Der Patentverletzungsprozeß, 4. Aufl. 1999, Köln
- Schuima, Daniele** TRIPS und das Patentierungsverbot von Software "als solcher" GRUR Int. 1998, S. 855 ff.
- Schwarz van Beck, Philip** Der Zugang zu wesentlichen Einrichtungen nach europäischem und deutschem Kartellrecht – Eine rechtsvergleichende Untersuchung der Essential Facility-Doktrin im Rahmen von Art. 82 EGV und § 19 Abs. 4 Nr. 4 GWB, Berlin 2003
- Sedlmaier, Roman/ Gigerich, Jan** Rechtliche Bedingungen und Risiken der Landeshauptstadt München für den Einsatz von Open-Source Software, 2004 (<http://www.ris-muenchen.de/RII/RII/DOK/SITZUNGSVORLAGE/517379.pdf>, letzter Abruf: 16. November 2004)
- Siepmann, Jürgen** Softwarepatente/ Open Source – Öffentliches Expertengespräch des Ausschusses für Kultur und Medien sowie des Rechtsausschusses des Deutschen Bundestages 2001 (http://bundestag.de/gremien/welt/glob_end/v12_literatur_lm.html; letzter Abruf 2. Oktober 2004)
- Stallings, William** Betriebssysteme - Prinzipien und Umsetzung, Pearson Studium, München 2003
- Tanenbaum, Andrew** Computernetzwerke, Pearson Studium, München 2003
- Tanenbaum Andrew/van Stehen, Marten** Verteilte Systeme - Grundlagen und Paradigmen, Pearson Studium, München 2003
- Tauchert, Wolfgang** Zum Begriff der technischen Erfindung, JurPC Web-Dok. 28/2002
- ders.** Nochmals: Anforderungen an einen Patentschutz für Computerprogramme, GRUR 2004, S. 922 - 923

- Ullrich, Hanns/ Körner, Eberhard (Hrsg.)** Der internationale Softwarevertrag, Heidelberg 1995 (zit.: Bearbeiter in: Ullrich/Körner, Teil ... Rdnr. ...).
- Walter, Michel (Hrsg.)** Europäisches Urheberrecht, Berlin 2001 (zit.: Bearbeiter in: Europäisches Urheberrecht, Art. .. Rdnr. ...)
- Weyand, Joachim/ Haase, Heiko** Anforderungen an einen Patentschutz für Computerprogramme, GRUR 2004, S. 198 ff.
- Williamson, Siehnaï** The International Enforcement of Software Copyrights and Patens, Computer Law Review and Technology Journal 1998, S. 63 ff.
- Winischhofer, Thomas** Computersoftware und Patentrecht, Diss. 2000 (<http://www.winischhofer.net/other/patent.pdf> letzter Abruf 5. Oktober 2004)
- Wolf, Gunnar** Kartellrechtliche Grenzen von Produktinnovationen – Lehren aus den Verfahren gegen IBM und Microsoft für die Anwendung des Kartellrechts in Hochtechnologiemärkten, Baden-Baden 2004

Abkürzungsverzeichnis, Glossar

Debugger	Spezielles Computerprogramm zur Diagnose und Fehlerbehebung von Programmen
BAPI	Business Application Programming Interface Mit Hilfe von BAPI kann man im SAP R3 Business Objekte mittels RFC-Protokoll direkt manipulieren.
DEC-NET	Eine der ersten Peer-to-Peer Netzwerk Architekturen, entwickelt von der Digital Equipment Corporation
OSI	Open Systems Interconnection Ein Konzept zur Schaffung einer gemeinsamen Basis zur Datenkommunikation. Das OSI-Referenzmodell ist ein offenes Schichtenmodell, das die Netzwerkkommunikation in 7 Schichten unterteilt.
TCP	Transmission Control Protocol
UDP	User Datagram Protocol TCP und UDP sind auf IP aufsetzende Transportprotokolle der Schicht 4. Während UDP verbindungslos ist, stellt TCP eine sichere Verbindung bereit.
ISDN	Integrated Services Digital Network Standard für ein digitales, leitungsvermitteltes Telefonnetz.
Ethernet	Ethernet ist eine Technologie zur Verbindung von Computersystemen in lokalen Netzwerken. Es beschreibt Kabeltypen, Signalisierung und Protokolle.
MAC-Adressen	Media Access Control Adresse Die MAC Adresse ist eine eindeutige Adresse, die jedes Computernetzwerkgerät identifiziert.
DICOM	Digital Imaging and Communications in Medicine DICOM ist ein offener Standard zum Austausch von Informationen. Er wird häufig zum Austausch von Bildern in der Medizin verwendet.
IBM-SNA	IBM Systems Networking Architecture Eine in den 70er Jahren von IBM entwickelte Netzwerkarchitektur.
EBCDIC	Extended Binary Coded Decimals Interchange Code Von IBM entwickelter 8 bit Code, der ausschließlich auf IBM Großrechnern Verwendung findet.
ASCII	American Standard Code for Information Interchange Zeichensatz, der auf dem lateinischen Alphabet basiert.

EDIFACT	Electronic Data Interchange For Administration, Commerce and Transport Dieser von der UN definierte Standard dient dem elektronischen Austausch von Handelsdokumenten und Geschäftsnachrichten.
XML	Extensible Markup Language Standard zur Definition von Auszeichnungssprachen, die zum Datenaustausch oder zur Darstellung von Daten dienen.
SMTP	Simple Mail Transfer Protocol Protokoll, das den Versand von eMail in TCP/IP Netzwerken regelt.
FTP	File Transfer Protocol Protokoll, das zum Austausch von Dateien in TCP/IP Netzwerken verwendet wird.
HTTP	Hypertext Transfer Protocol Zustandsloses Protokoll zum Austausch von Dateien, wird hauptsächlich zur Übertragung von HTML Dateien im World Wide Web verwendet.
TCP/IP	Transmission Control Protocol / Internet Protocol Familie von Netzwerkprotokollen, die als Grundlage für die Kommunikation im Internet dienen.
HL7	Health Level Seven Internationaler Standard zum Austausch medizinischer Daten. Die Darstellung erfolgt im ASCII-Format. Die Umsetzung des Standards ist meist herstellerabhängig.
Ubiquitous Computing	Bezeichnet die Allgegenwärtigkeit der Informationsverarbeitung. (lat. ubique = überall)
Pervasive Computing	Bezeichnet die Durchdringung des Alltags mit „intelligenten,, Gegenständen (lat. pervadere = Durchdringen)