



# BUNDESPATENTGERICHT

IM NAMEN DES VOLKES

URTEIL

Verkündet am  
5. Dezember 2013

2 Ni 9/12 (EP)

---

(Aktenzeichen)

...

In der Patentnichtigkeitsache

...

**betreffend das europäische Patent 0 618 540**

**(DE 694 29 378)**

hat der 2. Senat (Nichtigkeitssenat) des Bundespatentgerichts auf Grund der mündlichen Verhandlung vom 05. Dezember 2013 unter Mitwirkung der Vorsitzenden Richterin Sredl, der Richter Merzbach und Dipl.-Ing. Baumgardt, der Richterin Dipl.-Phys. Dr. Thum-Rung sowie des Richters Dipl.-Phys. Dr. Forkel

für Recht erkannt:

- I. Das europäische Patent 0 618 540 wird mit Wirkung für das Hoheitsgebiet der Bundesrepublik Deutschland für nichtig erklärt.
- II. Die Kosten des Rechtsstreits trägt die Beklagte.
- III. Das Urteil ist gegen Sicherheitsleistung in Höhe von 120 % des zu vollstreckenden Betrages vorläufig vollstreckbar.

**Tatbestand**

Die Beklagte ist eingetragene Inhaberin des am 31. März 1994 unter Inanspruchnahme einer US-Priorität (US 41 497) vom 1. April 1993 in englischer Sprache angemeldeten und mit Datum 12. Dezember 2001 veröffentlichten europäischen Patents EP 0 618 540 B1 mit der Bezeichnung „Common name space for long and short filenames“, welches vom Deutschen Patent- und Markenamt unter der Nummer DE 694 29 378 mit der deutschen Bezeichnung „Gemeinsamer Speicherbereich für lange und kurze Dateinamen“ geführt wird.

Das Streitpatent war bereits Gegenstand eines Nichtigkeitsverfahrens vor dem Bundespatentgericht (2 Ni 2/05 EU). Das Bundespatentgericht hat das Streitpatent mit Urteil vom 26. Oktober 2006 mit Wirkung für das Hoheitsgebiet der Bundesrepublik Deutschland für nichtig erklärt. Auf die Berufung der Patentinhaberin hat der Bundesgerichtshof mit Urteil vom 20. April 2010 (X ZR 27/07) das vorgenannte Urteil des Bundespatentgerichts abgeändert und die Nichtigkeitsklage im Verfahren 2 Ni 2/05 abgewiesen.

Das Streitpatent umfasst 23 Patentansprüche. Der Patentanspruch 1 ist auf ein „Verfahren zum Betreiben eines Datenverarbeitungssystems, das einen Speicher, der ein Betriebssystem enthält, sowie einen Prozessor, der das Betriebssystem ausführt, umfasst“, gerichtet. Der ihm nebengeordnete Patentanspruch 12 ist auf das zugehörige „Datenverarbeitungssystem“ gerichtet, der weitere nebengeordnete Patentanspruch 23 auf ein entsprechendes „Computerlesbares Medium“ mit Anweisungen, um das Verfahren nach einem der Ansprüche 1 bis 11 auszuführen, gerichtet. Die Patentansprüche 2 bis 11 und 13 bis 22 sind Unteransprüche.

In der Verfahrenssprache Englisch haben die erteilten unabhängigen Patentansprüche 1, 12 und 23 folgenden Wortlaut:

1. A method of operating a data processing system (10) comprising memory (16) holding an operating system (17), and a processor (12) for running the operating system (17), the method comprising:

(a) storing (58, 59) in the memory (16) a first directory entry (18) holding a short filename for a file;

(b) storing (58, 59) in the memory (16) a second directory entry (20) being associated with the first directory entry (18) and holding a long filename for the file, said long filename having more characters than said short filename, said second directory entry (20) further holding information (42) indi-

cating that said second directory entry (20) holds said long filename; and

(c) in case that the operating system (17) permits only short filenames and said information (42) is set to make said second directory entry (20) invisible to the operating system (17), locating the file by accessing said first directory entry (18) or, in case that the operating system (17) permits long filenames and said information (42) is set to make said second directory entry (20) visible to the operating system (17), locating the file by accessing said second directory entry (20).

**12.** A data processing system (10), comprising:

(a) memory (16) holding:

- (i) an operating system (17),
- (ii) a first directory entry (18) holding a short filename for a file, and
- (iii) a second directory entry (20) being associated with the first directory entry (18) and holding a long filename for the file, said long filename having more characters than said short filename, said second directory entry (20) further holding information (42) indicating that said second directory entry (20) holds said long filename; and

(b) a processor (12) for running the operating system (17) and, in case that the operating system (17) permits only short filenames and said information (42) is set to make said second directory entry (20) invisible to the operating system (17), locating the file by accessing said first directory entry (18) or, in case

that the operating system (17) permits long filenames and said information (42) is set to make said second directory entry (20) visible to the operating system (17), locating the file by accessing said second directory entry (20).

**23.** A computer-readable medium having computer-executable instructions adapted to enable a data processing system to perform the method of one of claims 1 to 11.

Hinsichtlich des Wortlauts der auf den erteilten Patentanspruch 1 direkt oder indirekt rückbezogenen Unteransprüche 2 bis 11 sowie der auf den erteilten Patentanspruch 12 direkt oder indirekt rückbezogenen Unteransprüche 13 bis 22 wird auf die Streitpatentschrift Bezug genommen.

In der deutschen Übersetzung lauten die Ansprüche 1, 12 und 23:

**1.** Verfahren zum Betreiben eines Datenverarbeitungssystems (10), das einen Speicher (16), der ein Betriebssystem (17) enthält, sowie einen Prozessor (12), der das Betriebssystem (17) ausführt, umfasst, wobei das Verfahren umfasst:

a) Speichern (58, 59) eines ersten Verzeichniseintrags (18), der einen kurzen Dateinamen für eine Datei enthält, in dem Speicher (16);

b) Speichern (58, 59) eines zweiten Verzeichniseintrags (20), der mit dem ersten Verzeichniseintrag (18) verknüpft ist und einen langen Dateinamen für die Datei enthält, in dem Speicher (16), wobei der lange Dateiname mehr Zeichen hat als der kurze Dateiname und der zweite Verzeichniseintrag (20) des Weiteren Informationen (42) enthält, die anzeigen, dass

der zweite Verzeichniseintrag (20) den langen Dateinamen enthält; und

c) wenn das Betriebssystem (17) nur kurze Dateinamen zulässt und die Informationen (42) so eingestellt sind, dass der zweite Verzeichniseintrag (20) für das Betriebssystem (17) unsichtbar ist, Auffinden der Datei durch Zugreifen auf den ersten Verzeichniseintrag (18), oder, wenn das Betriebssystem (17) lange Dateinamen zulässt und die Informationen (42) so eingestellt sind, dass der zweite Verzeichniseintrag (20) für das Betriebssystem (17) sichtbar ist, Auffinden der Datei durch Zugreifen auf den zweiten Verzeichniseintrag (20).

**12. Datenverarbeitungssystem (10), das umfasst:**

(a) einen Speicher (16), der enthält:

1. ein Betriebssystem (17),
2. einen ersten Verzeichniseintrag (18), der einen kurzen Dateinamen für eine Datei enthält,
3. einen zweiten Verzeichniseintrag (20), der mit dem ersten Verzeichniseintrag (18) verknüpft ist und einen langen Dateinamen für die Datei enthält, wobei der lange Dateiname mehr Zeichen hat als der kurze Dateiname und der zweite Verzeichniseintrag (20) des Weiteren Informationen (42) enthält, die anzeigen, dass der zweite Verzeichniseintrag (20) den langen Dateinamen enthält; und

(b) einen Prozessor (12), der das Betriebssystem (17) ausführt und, wenn das Betriebssystem (17) nur kurze Dateinamen zulässt und die Informationen (42) so eingestellt sind, dass der zweite Verzeichniseintrag (20) für das Betriebssystem (17) unsichtbar ist, die Datei durch Zugreifen auf den ersten Verzeichniseintrag (18) auffindet, oder, wenn das Betriebssystem (17) lange Dateinamen zulässt und die Informationen (42) so eingestellt sind, dass der zweite Verzeichniseintrag (20) für das Betriebssystem (17) sichtbar ist, die Datei durch Zugreifen auf den zweiten Verzeichniseintrag (20) auffindet.

**23.** Computerlesbares Medium mit von einem Computer ausführbaren Anweisungen, die ein Datenverarbeitungssystem in die Lage versetzen, das Verfahren nach einem der Ansprüche 1 bis 11 auszuführen.

Die Klägerin macht geltend, der Gegenstand des Streitpatents sei gegenüber dem Stand der Technik nicht patentfähig. Er sei nicht neu, beruhe aber jedenfalls nicht auf einer erfinderischen Tätigkeit (Artikel II § 6 Absatz 1 Nr. 1 IntPatÜG, Artikel 138 Abs. 1 Buchst. a) EPÜ i. V. m. Artikel 54 Absatz 1, 2 und Artikel 56 EPÜ). Dazu verweist die Klägerin auf die vorveröffentlichten Druckschriften

**D1:** Beitrag von Linus Torvalds vom 23. Dezember 1992 in der Usenet-Newsgruppe comp.os.minix

**D2:** Beitrag von Georg Wallmann (unter seinem Nutzernamen „Natuerlich!“) vom 24. März 1992 in der Usenet-Newsgruppe comp.sys.atari.st.tech

sowie auf die in der mündlichen Verhandlung vorgelegten Druckschriften und Dokumente

**D3:** US 5 083 264

**D4:** US 5 307 494

und macht geltend, dass der Gegenstand des geltenden Anspruchs 1 bereits durch die technischen Lehren der Druckschriften D1 und D2 – welche weder in Erteilungsverfahren noch in dem vorgenannten Nichtigkeitsverfahren als Stand der Technik berücksichtigt worden seien - neuheitsschädlich vorweggenommen werde. Entsprechendes gelte für die jeweiligen Gegenstände nach den unabhängigen Patentansprüchen 12 und 23. Darüber hinaus seien dem Fachmann die jeweiligen Merkmale der restlichen Patentansprüche aus dem Stand der Technik zumindest nahegelegt.

Weiterhin macht sie geltend, dass es sich bei dem Gegenstand des Streitpatents um reine Software, also um „Programme für Datenverarbeitungsanlagen als solche“ handele, welche dem Ausschlussstatbestand des Art. 52 Abs. 2 c) und Abs. 3 EPÜ unterfalle.

Die Klägerin beantragt,

das europäische Patent EP 0 618 540 mit Wirkung für das Hoheitsgebiet der Bundesrepublik Deutschland für nichtig zu erklären.

Die Beklagte beantragt,

die Klage abzuweisen;

hilfsweise beantragt sie, dem Streitpatent eine der Fassungen der Hilfsanträge 1 bis 5, vorgelegt in der mündlichen Verhandlung, zu geben.



### **Hilfsantrag 1**

Die unabhängigen Patentansprüche 1 und 11 in der Fassung des **Hilfsantrags 1** lauten (mit markierten Änderungen gegenüber den Patentansprüchen 1 und 12 in der erteilten Fassung):

1. A method of operating a data processing system (10) comprising memory (16) holding an operating system (17), and a processor (12) for running the operating system (17), the method comprising:

(a) storing (58, 59) in the memory (16) a first directory entry (18) holding a short filename for a file;

(b) storing (58, 59) in the memory (16) a second directory entry (20) being associated with the first directory entry (18) and holding a long filename for the file, said long filename having more characters than said short filename, said second directory entry (20) further holding information (42) indicating that said second directory entry (20) holds said long filename; and

(c) in case that the operating system (17) permits only short filenames and said information (42) is set to make said second directory entry (20) invisible to the operating system (17), locating the file by accessing said first directory entry (18) or, in case that the operating system (17) permits long filenames and said information (42) is set to make said second directory entry (20) visible to the operating system (17), locating the file by accessing said second directory entry (20);

wherein the step of storing (58, 59) said second directory

entry (20) further comprises:

storing a checksum of said short filename in said second directory entry (20).

11. A data processing system (10), comprising:

(a) memory (16) holding:

- (i) an operating system (17),
- (ii) a first directory entry (18) holding a short filename for a file, and
- (iii) a second directory entry (20) being associated with the first directory entry (18) and holding a long filename for the file, said long filename having more characters than said short filename, said second directory entry (20) further holding information (42) indicating that said second directory entry (20) holds said long filename; ~~and~~

(b) a processor (12) for running the operating system (17) and, in case that the operating system (17) permits only short filenames and said information (42) is set to make said second directory entry (20) invisible to the operating system (17), locating the file by accessing said first directory entry (18) or, in case that the operating system (17) permits long filenames and said information (42) is set to make said second directory entry (20) visible to the operating system (17), locating the file by accessing said second directory entry (20); and

means (12) for storing a checksum of said short filename in said second directory entry (20).

---

Die weiteren Patentansprüche gemäß Hilfsantrag 1 entsprechen den erteilten Unteransprüchen 2 und 3 sowie den neu nummerierten und mit entsprechend geänderten Rückbezügen versehenen erteilten Unteransprüchen 5 bis 11 (nunmehr 4 bis 10), 13 und 14 (nunmehr 12 und 13), 16 bis 22 (nunmehr 14 bis 20) sowie dem nebengeordneten Patentanspruch 23 in der erteilten Fassung (nunmehr 21).

## **Hilfsantrag 2**

Die unabhängigen Patentansprüche 1 und 12 (nunmehr Patentanspruch 11) in der Fassung des **Hilfsantrags 2** lauten (mit markierten Änderungen gegenüber den Patentansprüchen 1 und 12 in der erteilten Fassung):

1. A method of operating a data processing system (10) comprising memory (16) holding an operating system (17), and a processor (12) for running the operating system (17), the method comprising:

(a) storing (58, 59) in the memory (16) a first directory entry (18) holding a short filename for a file;

(b) storing (58, 59) in the memory (16) a second directory entry (20) being associated with the first directory entry (18) and holding a long filename for the file, said long filename having more characters than said short filename, said second directory entry (20) further holding information (42) indicating that said second directory entry (20) holds said long filename; and

(c) in case that the operating system (17) permits only short filenames and said information (42) is set to make said second directory entry (20) invisible to the operating system (17), locating the file by accessing said first directory entry (18) or, in case that the operating system (17) permits long

filenames and said information (42) is set to make said second directory entry (20) visible to the operating system (17), locating the file by accessing said second directory entry (20);

wherein the method further comprises the steps of:

creating the file;

assigning the file said long filename that is user-specified;  
and

creating said short filename of fewer characters by manipulating said long filename.

11. A data processing system (10), comprising:

(a) memory (16) holding:

- (i) an operating system (17),
- (ii) a first directory entry (18) holding a short filename for a file, and
- (iii) a second directory entry (20) being associated with the first directory entry (18) and holding a long filename for the file, said long filename having more characters than said short filename, said second directory entry (20) further holding information (42) indicating that said second directory entry (20) holds said long filename; ~~and~~

(b) a processor (12) for running the operating system (17) and, in case that the operating system (17) permits only short filenames and said information (42) is set to make said second directory entry (20) invisible to the operating system (17), locating

the file by accessing said first directory entry (18) or, in case that the operating system (17) permits long filenames and said information (42) is set to make said second directory entry (20) visible to the operating system (17), locating the file by accessing said second directory entry (20);

(c) means (12) for creating the file;

(d) means (12) for assigning the file said long filename that is user-specified; and

(e) means (12) for creating said short filename of fewer characters by manipulating said long filename.

Die weiteren Patentansprüche gemäß Hilfsantrag 2 entsprechen den erteilten Unteransprüchen 2 bis 10, den neu nummerierten und mit entsprechend geänderten Rückbezügen versehenen erteilten Unteransprüchen 13 bis 21 (nunmehr 12 bis 20) sowie dem nebengeordneten Patentanspruch 23 in der erteilten Fassung (nunmehr 21).

### **Hilfsantrag 3**

Die unabhängigen Patentansprüche 1 und 11 in der Fassung des **Hilfsantrags 3** lauten (mit markierten Änderungen gegenüber den Patentansprüchen 1 und 11 nach Hilfsantrag 1):

1. A method of operating a data processing system (10) comprising memory (16) holding an operating system (17), and a processor (12) for running the operating system (17), the method comprising:

(a) storing (58, 59) in the memory (16) a first directory entry (18) holding a short filename for a file;

(b) storing (58, 59) in the memory (16) a second directory entry (20) being associated with the first directory entry (18) and holding a long filename for the file, said long filename having more characters than said short filename, said second directory entry (20) further holding information (42) indicating that said second directory entry (20) holds said long filename; and

(c) in case that the operating system (17) permits only short filenames and said information (42) is set to make said second directory entry (20) invisible to the operating system (17), locating the file by accessing said first directory entry (18) or, in case that the operating system (17) permits long filenames and said information (42) is set to make said second directory entry (20) visible to the operating system (17), locating the file by accessing said second directory entry (20);

wherein the step of storing (58, 59) said second directory entry (20) further comprises:

storing a checksum of said short filename in said second directory entry (20),

wherein said checksum is used to associate the second directory entry with the first directory entry.

---

11. A data processing system (10), comprising:

(a) memory (16) holding:

(i) an operating system (17),

(ii) a first directory entry (18) holding a short filename for a file, and

(iii) a second directory entry (20) being associated with the first directory entry (18) and holding a long filename for the file, said long filename having more characters than said short filename, said second directory entry (20) further holding information (42) indicating that said second directory entry (20) holds said long filename; ~~and~~

(b) a processor (12) for running the operating system (17) and, in case that the operating system (17) permits only short filenames and said information (42) is set to make said second directory entry (20) invisible to the operating system (17), locating the file by accessing said first directory entry (18) or, in case that the operating system (17) permits long filenames and said information (42) is set to make said second directory entry (20) visible to the operating system (17), locating the file by accessing said second directory entry (20); and

means (12) for storing a checksum of said short filename in said second directory entry (20),

wherein said checksum is used to associate the second directory entry with the first directory entry.

---

Die weiteren Patentansprüche gemäß Hilfsantrag 3 entsprechen den Unteransprüchen 2 bis 10 und 12 bis 20 sowie dem nebengeordneten Anspruch 21 gemäß Hilfsantrag 1.

#### **Hilfsantrag 4**

Die unabhängigen Patentansprüche 1 und 11 in der Fassung des **Hilfsantrags 4** lauten (mit markierten Änderungen gegenüber den Patentansprüchen 1 und 11 nach **Hilfsantrag 2**):

1. A method of operating a data processing system (10) comprising memory (16) holding an operating system (17), and a processor (12) for running the operating system (17), the method comprising:

(a) storing (58, 59) in the memory (16) a first directory entry (18) holding a short filename for a file;

(b) storing (58, 59) in the memory (16) a second directory entry (20) being associated with the first directory entry (18) and holding a long filename for the file, said long filename having more characters than said short filename, said second directory entry (20) further holding information (42) indicating that said second directory entry (20) holds said long filename; and

(c) in case that the operating system (17) permits only short filenames and said information (42) is set to make said second directory entry (20) invisible to the operating system (17), locating the file by accessing said first directory entry (18) or, in case that the operating system (17) permits long filenames and said information (42) is set to make said second directory entry (20) visible to the operating system (17),



locating the file by accessing said second directory entry (20);

wherein the method further comprises the steps of:

creating the file;

assigning the file said long filename that is user-specified;  
and

creating said short filename of fewer characters by ~~manipulating~~ shortening said long filename so that the short filename does not exceed eight characters, absent the extension.

11. A data processing system (10), comprising:

(a) memory (16) holding:

(i) an operating system (17),

(ii) a first directory entry (18) holding a short filename for a file, and

(iii) a second directory entry (20) being associated with the first directory entry (18) and holding a long filename for the file, said long filename having more characters than said short filename, said second directory entry (20) further holding information (42) indicating that said second directory entry (20) holds said long filename; ~~and~~

(b) a processor (12) for running the operating system (17) and, in case that the operating system (17) permits only short filenames and said information (42) is set to make said second directory entry (20) invisible to the operating system (17), locating

the file by accessing said first directory entry (18) or, in case that the operating system (17) permits long filenames and said information (42) is set to make said second directory entry (20) visible to the operating system (17), locating the file by accessing said second directory entry (20);

(c) means (12) for creating the file;

(d) means (12) for assigning the file said long filename that is user-specified; and

(e) means (12) for creating said short filename of fewer characters by manipulating shortening said long filename so that the short filename does not exceed eight characters, absent the extension.

Die weiteren Patentansprüche gemäß Hilfsantrag 4 entsprechen den Unteransprüchen 2 bis 10 und 12 bis 20 sowie dem nebengeordneten Anspruch 21 gemäß Hilfsantrag 2.

### **Hilfsantrag 5**

Die unabhängigen Patentansprüche 1 und 11 in der Fassung des **Hilfsantrags 5** lauten (mit markierten Änderungen gegenüber den Patentansprüchen 1 und 11 nach Hilfsantrag 4):

1. A method of operating a data processing system (10) comprising memory (16) holding an operating system (17), and a processor (12) for running the operating system (17), the method comprising:

(a) storing (58, 59) in the memory (16) a first directory entry (18) holding a short filename for a file;

(b) storing (58, 59) in the memory (16) a second directory entry (20) being associated with the first directory entry (18) and holding a long filename for the file, said long filename having more characters than said short filename, said second directory entry (20) further holding information (42) indicating that said second directory entry (20) holds said long filename; and

(c) in case that the operating system (17) permits only short filenames and said information (42) is set to make said second directory entry (20) invisible to the operating system (17), locating the file by accessing said first directory entry (18) or, in case that the operating system (17) permits long filenames and said information (42) is set to make said second directory entry (20) visible to the operating system (17), locating the file by accessing said second directory entry (20);

wherein the method further comprises the steps of:

creating the file;

assigning the file said long filename that is user-specified;  
and

creating said short filename of fewer characters by shortening said long filename so that the short filename does not exceed eight characters, absent the extension, and does not collide with any other short filename.

**11.** A data processing system (10), comprising:

(a) memory (16) holding:

- (i) an operating system (17),
- (ii) a first directory entry (18) holding a short filename for a file, and
- (iii) a second directory entry (20) being associated with the first directory entry (18) and holding a long filename for the file, said long filename having more characters than said short filename, said second directory entry (20) further holding information (42) indicating that said second directory entry (20) holds said long filename; ~~and~~

(b) a processor (12) for running the operating system (17) and, in case that the operating system (17) permits only short filenames and said information (42) is set to make said second directory entry (20) invisible to the operating system (17), locating the file by accessing said first directory entry (18) or, in case that the operating system (17) permits long filenames and said information (42) is set to make said second directory entry (20) visible to the operating system (17), locating the file by accessing said second directory entry (20);

(c) means (12) for creating the file;

(d) means (12) for assigning the file said long filename that is user-specified; and

(e) means (12) for creating said short filename of fewer characters by shortening said long filename so that the short filename does not exceed eight characters, absent the extension, and does not collide with any other short filename.

Die weiteren Patentansprüche gemäß **Hilfsantrag 5** entsprechen den Unteransprüchen 2 bis 10 und 12 bis 20 sowie dem nebengeordneten Anspruch 21 gemäß **Hilfsantrag 4**.

Die Beklagte tritt den Ausführungen der Klägerin in allen Punkten entgegen. Das Streitpatent unterliege keinem Patentierungsausschluss. Das Streitpatent sei auch im Übrigen patentfähig, jedenfalls in einer der Fassungen der Hilfsanträge. Es mangle ihm gegenüber dem Stand der Technik weder an Neuheit noch an erfinderischer Tätigkeit.

Wegen der weiteren Einzelheiten des Vorbringens der Beteiligten wird auf den Akteninhalt verwiesen.

### **Entscheidungsgründe:**

Die Klage, mit der der Nichtigkeitsgrund der fehlenden Patentfähigkeit nach Artikel II § 6 Absatz 1 Nr. 1 IntPatÜG, Artikel 138 Abs. 1 lit a EPÜ i. V. m. Artikel 54 Absatz 1, 2 und Artikel 56 EPÜ geltend gemacht wird, ist zulässig. Sie ist auch begründet. Dabei bedarf die Frage, ob die Lehre des Streitpatents nach Art. 52 Abs. 2 Buchst. c oder d EPÜ vom Patentschutz ausgeschlossen ist, keiner abschließenden Entscheidung. Denn das Streitpatent erweist sich sowohl in der erteilten Fassung als auch in der Fassung der Hilfsanträge nicht als patentfähig, da die darin beanspruchte Lehre für den Fachmann durch den Stand der Technik nahegelegt ist (Artikel II § 6 Absatz 1 Nr. 1 IntPatÜG, Artikel 138 Abs. 1 lit a EPÜ i. V. m. Artikel 56 EPÜ).

#### **I.**

Die in der mündlichen Verhandlungen von der Beklagten vorgelegten Hilfsanträge waren trotz der Rüge der Klägerin nicht als verspätet zurückzuweisen.

Die durch das 2009 in Kraft getretene Patentrechtsmodernisierungsgesetz (PatR-ModG) erfolgte Neufassung des § 83 PatG und die damit in das Nichtigkeitsverfahren eingeführten Präklusionsregeln sehen zwar grundsätzlich die Möglichkeit vor, verspätetes Vorbringen zurückzuweisen. Hierfür ist es aber stets erforderlich, dass dieser Vortrag tatsächliche oder rechtliche Fragen aufkommen lässt, die in der mündlichen Verhandlung nicht oder nur mit unverhältnismäßigem Aufwand zu klären sind (vgl. Begründung zum Entwurf eines Gesetzes zur Vereinfachung und Modernisierung des Patentrechts, BIPMZ 2009, 307, 315). Kann das an sich verspätete Vorbringen dagegen noch ohne Weiteres in die mündliche Verhandlung einbezogen werden, ohne dass es zu einer Verfahrensverzögerung kommt, liegen die Voraussetzungen für eine Zurückweisung nach § 83 Abs. 4 PatG nicht vor. So liegt der Fall hier, weil das Streitpatent auch in den beschränkt verteidigten Anspruchsfassungen nach den Hilfsanträgen für nichtig zu erklären ist und die Berücksichtigung dieser Hilfsanträge, zu denen die Parteien verhandelt haben, auch zu keiner Verzögerung des Rechtsstreits geführt hat.

## II.

1. Ausweislich der erteilten Beschreibung betrifft das Streitpatent einen gemeinsamen Speicherbereich für kurze und lange Dateinamen. In der Beschreibungseinleitung wird erläutert, dass viele Betriebssysteme wie etwa MS-DOS, Version 5, nur kurze Dateinamen unterstützen. So könnten unter dem Betriebssystem MS-DOS, Version 5, Dateinamen höchstens elf Zeichen umfassen. Jeder Dateiname könne einen Hauptteil von bis zu acht Zeichen aufweisen, auf den eine Erweiterung von drei Zeichen folge. Ein Dateiname im Betriebssystem MS-DOS, Version 5, laute beispielsweise „EXAMPLE1.EXE“, wobei „EXAMPLE1“ der Hauptteil und „EXE“ die Erweiterung sei.

Jeder Dateiname sei aufgrund von Beschränkungen des Dateisystems im Betriebssystem MS-DOS, Version 5, auf elf Zeichen beschränkt. Eine derartige Beschränkung der Länge von Dateinamen sei jedoch unpraktisch für den Benutzer.

Die Längenbeschränkung auf elf Zeichen verhindere, dass der Benutzer adäquat beschreibende Dateinamen verwenden könne, und zwingt ihn in vielen Fällen, unschöne Abkürzungen von beschreibenden Namen als Dateinamen zu verwenden.

2. Davon ausgehend ist es Aufgabe der Erfindung, ein Verfahren und ein System anzugeben, die lange Dateinamen unterstützen (Streitpatentschrift, Absatz [0006]). Dabei soll das beanspruchte Verfahren bzw. System auch weiterhin kurze Dateinamen verwalten können, d. h. kompatibel mit solchen Betriebssystemen sein, die nur kurze Dateinamen für eine Datei verwenden können (Streitpatentschrift, Absätze [0015], [0016]).

3. Die genannte Aufgabe wird durch das Verfahren nach Patentanspruch 1 gelöst, welcher in der deutschen Übersetzung folgenden Wortlaut besitzt (unter weitgehender Berücksichtigung der von der Klägerin vorgeschlagenen Merkmalsgliederung):

1. „Verfahren zum Betreiben eines Datenverarbeitungssystems (10), das einen Speicher (16), der ein Betriebssystem (17) enthält, sowie einen Prozessor (12), der das Betriebssystem (17) ausführt, umfasst, wobei das Verfahren umfasst:
  - 1.1 Speichern (58, 59) eines ersten Verzeichniseintrags (18), der einen kurzen Dateinamen für eine Datei enthält, in dem Speicher (16);
  - 1.2 Speichern (58, 59) eines zweiten Verzeichniseintrags (20), der mit dem ersten Verzeichniseintrag (18) verknüpft ist und einen langen Dateinamen für die Datei enthält, in dem Speicher (16),
    - 1.2.1 wobei der lange Dateiname mehr Zeichen hat als der kurze Dateiname und

- 1.2.2 der zweite Verzeichniseintrag (20) des Weiteren Informationen (42) enthält, die anzeigen, dass der zweite Verzeichniseintrag (20) den langen Dateinamen enthält; und
- 1.3 wenn das Betriebssystem (17) nur kurze Dateinamen zulässt und die Informationen (42) so eingestellt sind, dass der zweite Verzeichniseintrag (20) für das Betriebssystem (17) unsichtbar ist, Auffinden der Datei durch Zugreifen auf den ersten Verzeichniseintrag (18),  
oder,
- 1.4 wenn das Betriebssystem (17) lange Dateinamen zulässt und die Informationen (42) so eingestellt sind, dass der zweite Verzeichniseintrag (20) für das Betriebssystem (17) sichtbar ist, Auffinden der Datei durch Zugreifen auf den zweiten Verzeichniseintrag (20).“

Weiterhin wird diese Aufgabe gelöst durch das System nach dem unabhängigen Patentanspruch 12, welcher sich wie folgend gliedern lässt und in der deutschen Fassung lautet:

1. „Datenverarbeitungssystem (10), das umfasst:
  - 1.1 einen Speicher (16), der enthält:
    - 1.1.1 ein Betriebssystem (17),
    - 1.1.2 einen ersten Verzeichniseintrag (18), der einen kurzen Dateinamen für eine Datei enthält,
    - 1.1.3 einen zweiten Verzeichniseintrag (20), der mit dem ersten Verzeichniseintrag (18) verknüpft ist und einen langen Dateinamen für die Datei enthält,
      - 1.1.3.1 wobei der lange Dateiname mehr Zeichen hat als der kurze Dateiname und



- 1.1.3.2. der zweite Verzeichniseintrag (20) des Weiteren Informationen (42) enthält, die anzeigen, dass der zweite Verzeichniseintrag (20) den langen Dateinamen enthält; und
- 1.2 einen Prozessor (12), der das Betriebssystem (17) ausführt und,
- 1.3 wenn das Betriebssystem (17) nur kurze Dateinamen zulässt und die Informationen (42) so eingestellt sind, dass der zweite Verzeichniseintrag (20) für das Betriebssystem (17) unsichtbar ist, die Datei durch Zugreifen auf den ersten Verzeichniseintrag (18) auffindet, oder,
- 1.4 wenn das Betriebssystem (17) lange Dateinamen zulässt und die Informationen (42) so eingestellt sind, dass der zweite Verzeichniseintrag (20) für das Betriebssystem (17) sichtbar ist, die Datei durch Zugreifen auf den zweiten Verzeichniseintrag (20) auffindet.”

4. Als zuständigen Fachmann hat der Bundesgerichtshof einen oder mehrere als Team zusammenarbeitende Informatiker oder Ingenieure mit Studienschwerpunkt Informatik angesehen, die über mehrjährige praktische Erfahrungen im Bereich der Systemprogrammierung verfügen.

5. Den erteilten Patentansprüchen 1, 12 und 23 lässt sich entsprechend der Beschreibung und den Figuren 1 bis 6b folgende Lehre entnehmen:

Das streitpatentgemäße Verfahren dient dem Betrieb eines Datenverarbeitungssystems, das einen Speicher und einen Prozessor umfasst, wobei der Speicher ein Betriebssystem beinhaltet, das von dem Prozessor ausgeführt wird (**Merkmal 1**).

Der Patentanspruch 12 betrifft ein entsprechendes Datenverarbeitungssystem, Patentanspruch 23 ein computerlesbares Medium. Da die beiden Patentansprüche inhaltlich nicht über den Patentanspruch 1 hinausgehen, wird nur der Patentanspruch 1 abgehandelt. Die Aussagen gelten entsprechend für die Patentansprüche 12 und 23.

Der Patentanspruch 1 geht davon aus, dass in dem Speicher für eine Datei ein erster und ein zweiter Verzeichniseintrag angelegt werden (**Merkmale 1.1 und 1.2**). Daraus ergibt sich, dass in dem Datenverarbeitungssystem, genauer Dateisystem, ein Verzeichnis angelegt ist, in dem alle Dateien mit zwei Dateinamen eingetragen sind, nämlich einem kurzen und einem langen Dateinamen. Die Verzeichniseinträge ermöglichen es, den Ort aufzufinden, an dem die jeweilige Datei abgespeichert ist.

Der erste Verzeichniseintrag enthält einen kurzen Namen und der zweite Verzeichniseintrag einen langen (aus mehr Zeichen als der kurze Name bestehenden) Namen für ein- und dieselbe Datei. Weist eine Datei einen Dateinamen von mehr als 11 Zeichen auf, so wird, um die Namensbeschränkung von MS-DOS zu umgehen, neben dem Verzeichniseintrag mit kurzem Dateinamen auch ein Verzeichniseintrag mit langem Dateinamen angelegt (Streitpatentschrift, Absätze [0020], [0021]).

Der erste Verzeichniseintrag ist mit dem zweiten Verzeichniseintrag verknüpft. Laut Beschreibung wird dies durch ein Prüfsummenfeld im zweiten Verzeichniseintrag erreicht, das eine Prüfsumme des kurzen Dateinamens enthält, sowie durch die Tatsache, dass der Verzeichniseintrag mit langem Dateinamen im Speicher direkt benachbart zum Verzeichniseintrag mit kurzem Dateinamen abgespeichert ist (Streitpatentschrift, Absätze [0021], [0031]).

Die Merkmale **1.2.1** und **1.2.2** besagen, dass der lange Dateiname über mehr Zeichen verfügt als der kurze Dateiname und dass der zweite Verzeichniseintrag In-

formationen (42) enthält, die anzeigen, dass der zweite Verzeichniseintrag den langen Dateinamen enthält.

Die **Merkmale 1.3** und **1.4** beschäftigen sich mit dem Auffinden einer Datei in Abhängigkeit vom jeweiligen Betriebssystem und von den bereits genannten Informationen (42). In **Merkmal 1.3**, der ersten Alternative, wird der Fall aufgegriffen, dass das Betriebssystem nur kurze Dateinamen zulässt und die Informationen (42) so eingestellt sind, dass der zweite Verzeichniseintrag für dieses Betriebssystem unsichtbar sind. Sind beide Bedingungen erfüllt, wird die Datei durch Zugreifen auf den ersten Verzeichniseintrag gefunden.

**Merkmal 1.4**, die zweite Alternative, besagt, dass die Datei durch Zugreifen auf den zweiten Verzeichniseintrag dann aufgefunden wird, wenn das Betriebssystem zum einen lange Dateinamen zulässt und zum anderen die Informationen (42) so eingestellt sind, damit der zweite Verzeichniseintrag sichtbar ist.

Laut Streitpatentschrift (Fig. 5a, 5b mit Beschreibung) sind die Informationen (42) einem Dateiattributefeld des zweiten Verzeichniseintrages entnehmbar. Das Dateiattributefeld verfügt hierfür über ein Hidden Bit, ein Read-only Bit, ein System Bit sowie ein Volume Label Bit. Den Ausführungen des Bundesgerichtshofs in seinem Urteil X ZR 27/07 zu demselben Streitpatent ist in diesem Kontext zu entnehmen, dass die Bitkombination „1111“ für ältere Betriebssysteme wie MS-DOS, die nur kurze Dateinamen verwalten können, ungültig ist. Für solche Betriebssysteme bleibt daher der zweite Verzeichniseintrag mit seinem langen Dateinamen verborgen bzw. unsichtbar. Auf der anderen Seite erkennen Betriebssysteme, die lange Dateinamen zulassen, aufgrund der Bitkombination „1111“, dass ein zweiter Verzeichniseintrag vorhanden ist. Für solche Betriebssysteme ist der zweite Verzeichniseintrag sichtbar, so dass die Datei durch diesen gefunden werden kann.

Alles in allem beruht die beanspruchte Lehre auf einem erweiterten Dateisystem, welches den Zugriff auf eine Datei sowohl über einen kurzen als auch langen Dateinamen ermöglicht, die in zwei verschiedenen Verzeichniseinträgen gespeichert

sind. Die zwei Verzeichniseinträge sind miteinander verknüpft, so dass die Datei sowohl unter Betriebssystemen mit als auch ohne Dateinamensbeschränkung aufgefunden werden kann.

## 6. Zum Hauptantrag

Das Streitpatent kann in der erteilten Fassung keinen Bestand haben, weil der Gegenstand des Patentanspruchs 1 bei Nicht-Berücksichtigung von Merkmalen, die zu einer technischen Problemlösung nicht beitragen, durch den Stand der Technik nahegelegt ist; die nebengeordneten Ansprüche und die Unteransprüche sind nicht anders zu beurteilen.

**6.1** Bei der Beurteilung der erfinderischen Tätigkeit sind nur solche Merkmale zu berücksichtigen, welche die Lösung eines technischen Problems, das durch zumindest einen Teilaspekt der unter Schutz gestellten Lehre bewältigt wird, mit technischen Mitteln bestimmen oder beeinflussen (BGH GRUR 2011, 125 – Wiedergabe topografischer Informationen).

Technische Mittel liegen vor, wenn Gerätekomponenten modifiziert oder grundsätzlich abweichend adressiert werden, wenn der Ablauf eines zur Problemlösung eingesetzten Datenverarbeitungsprogramms durch technische Gegebenheiten außerhalb der Datenverarbeitungsanlage bestimmt wird oder wenn die Lösung gerade darin besteht, ein Datenverarbeitungsprogramm so auszugestalten, dass es auf die technischen Gegebenheiten der Datenverarbeitungsanlage Rücksicht nimmt (BGH GRUR 2010, 613 – Dynamische Dokumentengenerierung; BGH GRUR 2011, 610 – Webseitenanzeige).

**6.2** Von besonderer Bedeutung ist die Druckschrift **D2** / „**Natuerlich!**“ (Comp.sys.atari.st.tech – Getting longer filenames out of GEMDOS).

Diese Druckschrift beschäftigt sich damit, wie unter dem Atari-Betriebssystem GEMDOS, welches eine Beschränkung von Dateinamen auf 8+3 Zeichen (Datei-

name+Extension) vorsieht, die Nutzung langer Dateinamen (mit mehr als 8 Zeichen im Dateinamen) ermöglicht werden kann.

In der Druckschrift **D2** wird im Wesentlichen vorgeschlagen, lange Dateinamen dadurch zu erhalten, dass eine Lösung aus mehreren Standard-GEMDOS-Verzeichniseinträgen aufgebaut wird. Die in der Druckschrift **D2** angeführten Pseudo-C Codefragmente dienen einem besseren Verständnis der vorgeschlagenen Verzeichniseinträge sowie deren weiterer Verarbeitung (Seite 1, Mitte – Seite 2, zweiter Absatz).

Die in der Druckschrift genannte Struktur *struct \_foo* der Programmiersprache C definiert einen eigenen neuen Datentyp, bestehend aus einer Sammlung von Daten, die nacheinander im Speicher stehen.

In der Druckschrift **D2** wird zuerst eine Struktur für den Aufbau eines 8.3 Verzeichniseintrages definiert, bestehend aus 8 Zeichen Name (*filename*), 3 Zeichen Extension (*ext*), *filetype* und 20 Zeichen *otherstuff*. Das Attribut *filetype* ist 1 Byte lang, so dass die gesamte Struktur eine Länge von 32 Bytes hat:

```
struct _foo  
{  
  char filename[8],  
  ext[3],  
  filetype,  
  otherstuff[20];  
}
```

Als nächstes wird eine Struktur für einen Verzeichniseintrag mit langem Dateinamen definiert, die zunächst wie die obige beginnt, dann zusätzlich ein Byte *unused* mit festem Wert *0xE5* enthält, und schließlich 31 Bytes für eine Erweiterung des Dateinamens. Somit ergibt sich eine Gesamtlänge von 64 Bytes für diese Struktur, die damit genau doppelt so lang ist wie jene für den kurzen Dateinamen.

Das Markierungsbyte *0xE5* kennzeichnet gelöschte Verzeichniseinträge. Es bewirkt bei älteren Betriebssystemen, welche nur Verzeichniseinträge mit der Länge 32 Byte kennen, dass der folgende Teil als eigener Verzeichniseintrag einer gelöschten Datei betrachtet und daher ignoriert wird. Wie nachfolgend beschrieben, erkennt ein für lange Dateinamen vorgesehenes Betriebssystem jedoch im vorhergehenden Eintrag an den Informationen in *filetype*, dass es sich um einen langen Dateinamen handelt, und greift auf die folgenden (mit *0xE5* beginnenden) 32 Byte als Erweiterung des vorherigen 32 Byte-Verzeichniseintrages zu. Der erweiterte Verzeichniseintrag wird durch die Struktur *struct \_baz* beschrieben:

```
struct _baz  
{  
    char filename[8],  
    ext[3],  
    filetype,  
    otherstuff[20];  
    unused = 0xE5;  
    longfilename[31];  
}
```

Nun werden beide Strukturen in einer *union \_bar* vereint. Die Komponenten einer *union* stehen nicht hintereinander im Speicher, sondern beginnen an der gleichen Adresse. Somit ist die Länge der gesamten *union* gleich der Länge der längsten Komponente, in diesem Fall *struct \_baz* mit Länge 64 Bytes. Die *union* bietet im Grunde nur unterschiedliche Zugriffsmechanismen, über *struct \_foo* nur auf den kurzen Namen, über *struct \_baz* auf den kurzen und langen Namen:

```
union _bar  
{  
    struct _foo;  
    struct _baz;  
}
```

Nachfolgend wird in der Druckschrift **D2** die Funktion zum Zugriff auf den erweiterten Dateinamen beschrieben, falls dieser gegeben ist. Hierzu werden für das Resultat eine Zeigervariable auf die Startadresse einer Zeichenkette sowie eine Zeigervariable für die Startadresse der *union* vom Typ *\_bar* angelegt:

```
extern char *filename;
```

```
_bar *p;
```

Weiterhin wird eine Konstante *LONGFILENAME* (hexadezimal *0x80* oder binär *1000 0000*) definiert. Wie am Ende der Druckschrift **D2** erwähnt, soll ein freies Bit des *filetypes* zur Unterscheidung verwendet werden, ob nur ein kurzer oder auch ein erweiterter Dateiname vorhanden ist (Seite 2, zweiter Absatz):

```
#define LONGFILENAME 0x80
```

Mittels Zeiger *p* wird auf die *union \_bar* und zwar deren Struktur *struct \_foo* und dort die Komponente *filename* zugegriffen. Der *&* Operator führt ein bitweises *AND* mit *0x80* durch, und das Ergebnis wird mittels *!* negiert. Die If-Abfrage wird dann wahr, wenn das Byte *filetype* im höchsten Bit *0* ist, d. h. wenn kein erweiterter Dateiname vorhanden ist:

```
if(!(p->_foo.filetype & LONGFILENAME))  
{  
    memcpy(filename, p->_foo.filename, 11);
```

Der *memcpy* Befehl kopiert Speicherbereiche gemäß *memcpy( ZielZeiger, UrsprungsZeiger, Anzahl Bytes)*. Wie im letzten Befehl wird auf die Zeichenkette *filename* in *struct \_foo* zugegriffen. Mit dem *memcpy* von 11 Bytes werden nicht nur der Dateiname *filename* (8 Bytes), sondern auch gleich die 3 Bytes Extension *ext* kopiert, da diese im Speicher direkt dahinter stehen. Danach erfolgt der Rücksprung aus der Funktion:

```
return;  
}
```

Andernfalls, d. h. wenn in *filetype* das höchste Bit 1 ist und somit ein erweiterter (langer) Dateiname vorliegt, wird mit den folgenden *memcpy* Befehlen von *filename*[8], *longfilename* [31] und *ext* [3] der lange Dateiname ermittelt:

```
memcpy (filename, p->_baz.filename,8);  
memcpy (filename, p->_baz.longfilename, 31);  
memcpy (filename, p->_baz.ext,3);
```

```
return;
```

Nach allem schlägt die Druckschrift **D2** im Wesentlichen Folgendes vor:

Kurze Dateinamen werden im Standard-GEMDOS-Verzeichniseintrag *struct \_foo* gespeichert, und zwar in den Feldern *filename* [8] und *ext* [3]. Lange Dateinamen werden über alle zur Verfügung stehenden Felder des erweiterten GEMDOS Verzeichniseintrags *struct \_baz* verteilt, nämlich *filename* [8], *ext* [3] und *longfilename*[31].

Um eine Abwärtskompatibilität zum Betriebssystem TOS der Heimcomputerserie Atari ST und MSDOS herzustellen, wird in der Druckschrift **D2** empfohlen, für diese Betriebssysteme den Eintrag *longfilename* per default mit *0xE5* zu markieren, damit dieser bei der Verzeichniserstellung bzw. -suche den älteren Betriebssystemen verborgen bleibt. Hierbei macht man sich die Tatsache zunutze, dass unter TOS und MSDOS mit *0xE5* gekennzeichnete Verzeichniseinträge bei der Erstellung einer Verzeichnisliste als gelöscht betrachtet und damit ignoriert werden.

Jedenfalls geht aus der Druckschrift **D2** hervor, dass für eine Datei mit langem Dateinamen ein erweiterter GEMDOS Verzeichniseintrag erstellt wird, welcher aus



zwei Standard GEMDOS Verzeichniseinträgen aufgebaut ist. Der lange Dateiname wird auf beide Verzeichniseinträge aufgeteilt.

Im Streitpatent sind demgegenüber für eine Datei mit langem Dateinamen immer ein erster Verzeichniseintrag, der einen zugehörigen kurzen Dateinamen speichert, und ein separater zweiter Verzeichniseintrag, der den (vollständigen) langen Dateinamen speichert, vorgesehen.

**6.3** Das Verfahren des Patentanspruchs 1 nach Hauptantrag beruht nicht auf erfinderischer Tätigkeit.

**6.3.1** Im Sinne der Lehre des Patentanspruchs 1 entnimmt der Fachmann der Druckschrift **D2** zunächst allgemein ein Verfahren zum Betreiben eines Datenverarbeitungssystems (Seite 1, erster Absatz, siehe „I will put out a few ideas how to get longer filenames out of GEMDOS...“; Seite 1, vierter Absatz, siehe „The most kludgy: in an effort to remain MSDOS compatible, how about this ...“). Die einzelnen Systemkomponenten des Datenverarbeitungssystems, wie etwa Speicher, Betriebssystem sowie Prozessor (auf dem das Betriebssystem ausgeführt wird) sind für den Fachmann in diesem Zusammenhang selbstverständlich und werden von ihm mitgelesen (**Merkmal 1**).

Die Druckschrift **D2** zeigt ihm einen erweiterten GEMDOS Verzeichniseintrag vom Datentyp *struct \_baz* mit einer Gesamtlänge von 64 Bytes. Dieser setzt sich aus zwei Standard GEMDOS Verzeichniseinträgen mit einer Länge von jeweils 32 Bytes zusammen (Seite 1, Mitte; siehe „one might consider allocating two successive entries in the directory table“).

Der erste der beiden gespeicherten Verzeichniseinträge umfasst u. a. die Felder *filename[8]* und *ext[3]*, welche Zeichenketten entsprechen und einen kurzen Dateinamen enthalten (Seite 1; siehe Struktur *struct \_baz* - **Merkmal 1.1**).

Der zweite der beiden gespeicherten Verzeichniseinträge wird gebildet aus dem Markierungsbyte *unused* sowie dem Feld *longfilename[31]* und speichert anstelle

eines vollständigen Dateinamens lediglich einen Teil eines langen Dateinamens. Gemäß der Struktur *struct \_baz* liegen die beiden Verzeichniseinträge im Speicher nebeneinander und sind auf diese Weise miteinander verknüpft (Seite 1; siehe Struktur *struct \_baz* - **teilweise Merkmal 1.2**).

Wie sich aus dem Feld *longfilename[31]* ergibt, sind im zweiten Verzeichniseintrag bis zu 31 Zeichen eines Teils des langen Dateinamens gespeichert. Demnach verfügt der lange Dateiname grundsätzlich über mehr Zeichen als der in den Feldern *filename[8]* und *ext[3]* des ersten Verzeichniseintrages gespeicherte kurze Dateiname (Seite 1; siehe Struktur *struct \_baz* - **Merkmal 1.2.1**).

Nach der speziellen Lehre des „pseudo access codes“ der Druckschrift **D2** (Seite 1, letzter Absatz – Seite 2, erster Absatz) wird die if-Bedingung falsch, wenn das Byte *filetype* im höchsten Bit 1 wird, d. h. wenn ein langer Dateiname vorliegt. In einem solchen Fall wird immer auf die Struktur *struct \_baz* zugegriffen, was anhand des *memcpy* Befehls auf Seite 2, oben veranschaulicht wird. Für ein Betriebssystem, welches nur kurze Dateinamen zulässt, also nur 8+3 Zeichen im Dateinamen, wird infolge des Markierungsbytes *0xE5* das Feld *longfilename[31]* und damit auch der zweite Verzeichniseintrag unsichtbar. Der Dateiname wird lediglich aus den Komponenten *\_baz.filename* und *\_baz.ext* des ersten Verzeichniseintrags zusammengefügt. Die Datei wird in einem solchen Fall also nur über einen Zugriff auf den ersten Verzeichniseintrag gefunden. Der Zugriffsmechanismus wird in der Druckschrift **D2** u. a. anhand der Anzeige („DIR search“) eines Verzeichnisses erläutert (Seite 2, erster und zweiter Absatz - **Merkmal 1.3**).

Lässt das Betriebssystem hingegen lange Dateinamen zu, wird auch die Komponente *longfilename* der Struktur *struct \_baz* sichtbar, und der Dateiname setzt sich zusammen aus den Komponenten des ersten Verzeichniseintrags, *\_baz.filename* und *\_baz.ext*, sowie aus der Komponente des zweiten Verzeichniseintrags *\_baz.longfilename*. Um eine Datei über den langen Dateinamen zu finden, wird dann nach der Lehre der Druckschrift **D2** (auch) auf den zweiten Verzeichniseintrag zugegriffen (Seite 2, erster Absatz - **Merkmal 1.4**).

Von der Lehre der Druckschrift **D2** unterscheidet sich die Lehre nach Patentanspruch 1 gemäß Hauptantrag nur noch dadurch, dass der zweite Verzeichniseintrag anstelle eines Teils eines langen Dateinamens einen vollständigen Dateinamen enthalten soll (restlicher Teil des **Merkmals 1.2**) und dass die Information für das Vorliegen eines langen Dateinamens, die sich gemäß Druckschrift **D2** aus dem Feld *filetype* im ersten Verzeichniseintrag ergibt, im zweiten Verzeichniseintrag abgelegt sein soll (**Merkmal 1.2.2**).

Um ausgehend von der Lehre der Druckschrift **D2** zum Gegenstand nach Patentanspruch 1 gemäß Hauptantrag zu gelangen, bedurfte es aus Sicht des zuständigen Fachmannes zum einen einer Umgruppierung von den Zeichenketten des auf die zwei Verzeichniseinträge der Struktur *struct \_baz* verteilten langen Dateinamens, so dass etwa die Komponente *longfilename[31]* nicht nur einen Teil des langen Dateinamens, sondern den vollständigen Dateinamen beinhaltet. Zum anderen bedurfte es einer Verschiebung des *filetype* Attributs vom ersten in den zweiten Verzeichniseintrag. Solche Maßnahmen bestehen in einer gedanklich logischen Anweisung zum Ordnen von Daten, die zwar gegenüber der Druckschrift **D2** den Zugriffsmechanismus auf den Dateinamen und damit auf die zugehörige Datei ändern, jedoch lediglich Aspekte der reinen Programmierung darstellen, denen keine auf technischen Überlegungen beruhenden Erkenntnisse zugrundeliegen. Solche Maßnahmen haben keine Lösung eines technischen Problems mit technischen Mitteln zum Gegenstand, und sind daher bei der Prüfung auf erfindetische Tätigkeit nicht zu berücksichtigen (BGH a. a. O. – Wiedergabe topografischer Informationen).

Sie enthalten weder einen Hinweis auf einen neuen Aufbau einer Datenverarbeitungsanlage noch auf irgendetwas, was über den „normalen“ bestimmungsgemäßen Gebrauch einer Datenverarbeitungsanlage hinausginge. Darüber hinaus liegen bei den zuletzt genannten Maßnahmen technische Mittel i. S. d. Entscheidung „Webseitenanzeige“ (BGH a. a. O.) nicht vor. Weder werden hierdurch Gerätekomponenten modifiziert, d. h. verändert oder grundsätzlich abweichend adressiert noch bewirken die Maßnahmen, dass das beanspruchte Verfahren durch

Gegebenheiten außerhalb des Datenverarbeitungssystems bestimmt wird, wie dies etwa bei einer gezielten Einstellung von Systemparametern der Fall ist. Insbesondere führen die Maßnahmen aber nicht dazu, dass auf die technischen Gegebenheiten des Datenverarbeitungssystems Rücksicht genommen wird, weil letztere an keiner Stelle des Streitpatents angesprochen werden.

Der Gegenstand des Patentanspruchs 1 gemäß Hauptantrag ist damit aus der Druckschrift **D2** nahegelegt.

**6.3.2** Demgegenüber wendet die Beklagte im Wesentlichen ein, dass sich der Bundesgerichtshof in seinem Urteil vom 20. April 2010 in der Patentnichtigkeitsache X ZR 27/07 bereits ausführlich mit der Frage des Ausschlussstatbestandes nach Artikel 52 (2) c) und (3) EPÜ auseinandergesetzt habe. Der Bundesgerichtshof sei zu dem Resultat gelangt, dass sich der Gegenstand des Streitpatents nicht auf ein Programm für Datenverarbeitungsanlagen als solches beziehe (Anlage NK8, Rn. 31 und 32, S. 17f). Die zum Zeitpunkt des Urteils bereits gefestigte Rechtsprechung des Bundesgerichtshofs zur Frage des Ausschlussstatbestands habe sich seit dem Urteil des Bundesgerichtshofs zum Streitpatent auch nicht geändert. Die unter Schutz gestellte Lösung betreffe „das technische Problem, wie bestimmte Daten in einem Speicher von Datenverarbeitungsanlagen zum Zugriff für unterschiedliche Betriebssysteme abgelegt werden müssen, und löst es mittels einer bestimmten Anordnung der Speicherbelegung“ (Anlage NK8, Rn. 32, S. 18). Der Bundesgerichtshof habe bestätigt, dass der Gegenstand des Streitpatents ein technisches Problem betreffe und dass das dieses Problem lösende Mittel eine bestimmte Anordnung der Speicherbelegung sei. Da alle Merkmale des Patentanspruchs 1 die Definition eben dieser Anordnung der Speicherbelegung beträfen, seien alle Merkmale des Patentanspruchs 1 technische Merkmale, die bei der Prüfung der erfinderischen Tätigkeit zu berücksichtigen seien.

Dem Einwand der Beklagten kann nicht gefolgt werden.

Die Frage, welche Merkmale des Streitpatentgegenstandes bei der Beurteilung der erfinderischen Tätigkeit Berücksichtigung finden bzw. die Frage, welche Merkmale es sind, die die Lösung eines technischen Problems mit technischen Mitteln zum Gegenstand haben, wurde im angeführten Urteil des Bundesgerichtshofs nicht entschieden. In der Entscheidung wurde lediglich festgestellt, dass dem Streitpatentgegenstand insgesamt eine technische Problemstellung zugrunde liegt (Anlage NK8, Rn. 32, S. 18; Rn. 41, S. 22 und 23). Weiterhin führte der Bundesgerichtshof aus, dass die wesentlichen Aspekte der streitpatentgemäßen Lehre in der Bitkombination „1111“ der Dateiattribute und im Anlegen von zwei Verzeichniseinträgen im Dateisystem zu sehen sind (Anlage NK8, Rn. 49, S. 26 und 27), die durch den damaligen Stand der Technik nicht nahegelegt waren. Gerade in der Druckschrift **D2** sind aber sowohl ein der Bitkombination „1111“ entsprechendes Datenfeld (Markierungsbyte *0xE5*) zur Maskierung eines Verzeichniseintrags als auch zwei Verzeichniseinträge vorhanden, deren Datenfelder gegenüber den Verzeichniseinträgen des Streitpatents nur auf andere Weise verteilt sind.

**6.3.3** Die Beklagte hält ferner entgegen, dass die in der Druckschrift **D2** definierte Struktur *struct \_baz* des langen Dateinamens nur einen einzigen erweiterten Verzeichniseintrag wiedergebe, mit dessen Hilfe eine Datei aufgefunden werden könne. Zwar würden zwei Datenstrukturen allokiert, jedoch gebe es keine Unterscheidung zweier Verzeichniseinträge. Mit dem in der Druckschrift **D2** gegebenen Hinweis „allocating two successive entries in the directory table“ seien nicht etwa zwei Verzeichniseinträge gemeint. Vielmehr werde durch die Struktur *struct \_baz* eine Datenstruktur der Größe 64 Bytes allokiert, die die doppelte Größe eines Standard GEMDOS Verzeichniseintrages besitze. Bei der Struktur *struct \_baz* verbleibe immer nur ein einziger Verzeichniseintrag, weswegen aus der Druckschrift **D2** auch keine Verknüpfung zweier Verzeichniseinträge ableitbar sei. Im Streitpatent gebe es hingegen zwei Verzeichniseinträge für ein und dieselbe Datei, die unabhängig voneinander seien.

Diese Sichtweise der Beklagten greift jedoch zu kurz. Nach fachmännischem Verständnis lehrt das Streitpatent, dass ein Verzeichniseintrag eine Datenstruktur mit

allen notwendigen Informationen darstellt, um eine Datei in einer Festplattenpartition wiederzufinden, insbesondere Dateiname, Disk Cluster, File Attribute usw. Die Informationen sind entweder selbst in der Datenstruktur enthalten oder werden durch Verknüpfung mit anderen Datenstrukturen eingebunden.

Solche Verzeichniseinträge sind auch in den Figuren 3a und 3b des Streitpatents weitgehend vollständig wiedergegeben, wobei der zweite Verzeichniseintrag über eine Verknüpfung Information einbindet, die im ersten Verzeichniseintrag hinterlegt ist (Streitpatentschrift, Absätze [0031]; [0041]). Der zweite Verzeichniseintrag ist z. B. mit dem Disk Cluster Feld des ersten Verzeichniseintrags verknüpft, da das eigene Cluster Feld auf Null gesetzt wird (Streitpatentschrift, Spalte 10, Zeilen 1-9). Die im Disk Cluster Feld enthaltene Nummer des Start Clusters der Datei stellt hierbei die für den Zugriff auf die Datei wohl wichtigste Information dar.

Die Verzeichniseinträge des Streitpatents sind nach allem nicht unabhängig voneinander, da der zweite Verzeichniseintrag zumindest einige der benötigten Datenfelder nicht selbst enthält, sondern aus dem ersten Verzeichniseintrag referenziert. Dies wird auch im Urteil X ZR 27/07 des Bundesgerichtshofs bestätigt, wonach die Verknüpfung des zweiten mit dem ersten Verzeichniseintrag ein Betriebssystem, das lange Dateinamen zulässt und Dateien durch Zugreifen auf den zweiten Verzeichniseintrag auffindet, in die Lage versetzt, auch auf Informationen zuzugreifen, die im ersten Verzeichniseintrag abgelegt sind (Anlage NK8, Rn. 21). Durch die Verknüpfung entfällt die Notwendigkeit, bestimmte die Dateien betreffende Informationen (redundant) in beiden Verzeichniseinträgen bereitzustellen.

In ähnlicher Weise ist in der Druckschrift **D2** die zweite Datenstruktur der Struktur *struct \_baz*, bestehend aus Markierungsbyte *unused* und *longfilename*, mit der ersten Datenstruktur verknüpft, und zwar in erster Linie, um die Komponenten *filename* und *ext* zur Bildung des langen Dateinamens zu nutzen.

Nach allem geht die Beklagte fehl, wenn sie beim Streitpatent von zwei unabhängigen Verzeichniseinträgen ausgeht. Vielmehr handelt es sich sowohl bei den

Verzeichniseinträgen des Streitpatents als auch der Druckschrift **D2** um jeweils zwei Datenstrukturen mit einer Größe von 32 Bytes, wobei der zweite Verzeichniseintrag (zweite Datenstruktur) auf Information des ersten Verzeichniseintrags (erste Datenstruktur) Bezug nimmt, wenn eine Datei mit langem Dateinamen gefunden werden soll.

Die Unterschiede zwischen den Verzeichniseinträgen des GEMDOS Dateisystems der Druckschrift **D2** und denjenigen des Streitpatents betreffen nur Maßnahmen aus der Welt der Programmierung, die keinen technischen Beitrag liefern können.

**6.3.4** Die Nebenansprüche 12 und 23 sind nicht anders zu bewerten als der Patentanspruch 1. Sie enthalten nichts Zusätzliches, womit sich eine eigenständige Patentfähigkeit begründen ließe.

Der auf ein Datenverarbeitungssystem gerichtete Patentanspruch 12 enthält zwar gegenüber dem Patentanspruch 1 zusätzlich ein paar technische Mittel (Speicher (16), Betriebssystem (17), Prozessor (12)), jedoch nichts, was im gegebenen Zusammenhang nicht selbstverständlich gewesen wäre. Daher ist für den Patentanspruch 12 eine andere Beurteilung als für den Hauptanspruch nicht gerechtfertigt.

Entsprechendes gilt für den auf ein computerlesbares Medium gerichteten Patentanspruch 23, der im Wesentlichen durch von einem Computer ausführbare Anweisungen zur Durchführung der Verfahrensschritte des Patentanspruchs 1 gekennzeichnet ist.

**6.3.5** Dass die zusätzlichen Merkmale der Unteransprüche, soweit sie nicht konkret Gegenstand von Hilfsanträgen sind, zu einer abweichenden Beurteilung der Patentfähigkeit führen könnten, wurde weder geltend gemacht, noch ist es sonst ersichtlich.

## 7. Zu den Hilfsanträgen 1 bis 5

Die Hilfsanträge 1 bis 5 können nicht anders beurteilt werden, weil keines der zusätzlichen Merkmale dem erteilten Patentanspruch 1 etwas hinzufügt, was die Patentfähigkeit tragen könnte. Die jeweils hinzugefügten Merkmale haben keine Lösung eines technischen Problems mit technischen Mitteln zum Gegenstand und sind daher bei der Beurteilung der erfinderischen Tätigkeit nicht zu berücksichtigen (BGH a. a. O. - Wiedergabe topografischer Informationen).

**7.1** Dem **Hilfsantrag 1** kann nicht stattgegeben werden, weil das neu hinzugekommene Merkmal zur Lösung eines technischen Problems mit technischen Mitteln nicht beitragen kann.

Gemäß Hilfsantrag 1 wird der Patentanspruch 1 nach Hauptantrag durch das Merkmal aus dem Unteranspruch 4 eingeschränkt, welches lautet:

„wobei der Schritt des Speicherns (58, 59) des zweiten Verzeichniseintrags (20) des Weiteren umfasst:

Speichern einer Prüfsumme des kurzen Dateinamens in dem zweiten Verzeichniseintrag (20).“

Entsprechend enthält der Nebenanspruch 11, der aus dem erteilten Patentanspruch 12 hervorgeht, zusätzlich das Merkmal

„(c) eine Einrichtung (12), die eine Prüfsumme des kurzen Dateinamens in dem zweiten Verzeichniseintrag (20) speichert.“

In den Absätzen [0031] und [0041] der Streitpatentschrift ist hierzu ausgeführt, dass der zweite Verzeichniseintrag ein „checksum field“ bzw. Prüfsummenfeld aufweist, das die Prüfsumme des kurzen Dateinamens angibt und dazu benutzt



wird, den langen Dateinamenseintrag mit dem entsprechenden kurzen Eintrag zu verknüpfen.

Dem Fachmann ist aber das Verwenden von Prüfsummen als Maßnahme zur Gewährleistung von Datenintegrität bei der Datenspeicherung aus seinem Fachwissen bekannt. Die Frage, auf welche Weise zwei Verzeichniseinträge miteinander verknüpft werden können, stellt kein technisches Problem dar, sondern ein Problem der Programmierung. In der Bildung von Prüfsummen, etwa durch eine einfache Aufsummierung von Datenwerten, und deren bloßer Speicherung in geeigneten Datenstrukturen können allenfalls Maßnahmen der reinen Datenverarbeitung erkannt werden (BGH a. a. O. – Webseitenanzeige), die aber keinerlei Lösung eines technischen Problems mit technischen Mitteln betreffen und deshalb bei der Beurteilung der erfinderischen Tätigkeit nicht zu berücksichtigen sind. Das hinzukommende Merkmal erlaubt keine Änderung der Beurteilung für den Patentanspruch 1.

Damit kann dahinstehen, ob – wie von der Klägerin vorgetragen – eine unzulässige Erweiterung vorliegt oder ob es sich bei dem neuen Merkmal um ein klares Merkmal handelt.

Mit dem Patentanspruch 1 fällt der gesamte Hilfsantrag. Beantragt der Patentinhaber, das Patent in beschränktem Umfang mit einem bestimmten Anspruchssatz aufrechtzuerhalten, rechtfertigt es grundsätzlich die Ablehnung des gesamten Antrages, wenn sich auch nur der Gegenstand eines Patentanspruchs aus dem vom Patentinhaber verteidigten Anspruchssatz als nicht patentfähig erweist (BGH GRUR 2007, 862 – Informationsübermittlungsverfahren II, bei Verteidigung des Patents durch Hilfsanträge auf das Nichtigkeitsverfahren zu übertragen).

**7.2 Der Hilfsantrag 2** kann nicht günstiger beurteilt werden, weil die neu hinzugekommenen Merkmale keine Lösung einer technischen Problemstellung zum Gegenstand haben.

Gemäß Hilfsantrag 2 wird der erteilte Patentanspruch 1 durch die folgenden Merkmale eingeschränkt

„wobei das Verfahren weiter die Schritte umfasst:

Erzeugen der Datei;

Zuordnen des langen Dateinamens, der vom Benutzer vorgegeben wird, zu der Datei; und

Erzeugen des kurzen Dateinamens aus weniger Zeichen durch Manipulieren des langen Dateinamens.“

Entsprechend enthält der Nebenanspruch 11, der aus dem erteilten Patentanspruch 12 hervorgeht, zusätzlich die Merkmale

„(c) eine Einrichtung (12), die die Datei erzeugt;

(d) eine Einrichtung (12), die den langen Dateinamen, der vom Benutzer vorgegeben wird, der Datei zuweist; und

(e) eine Einrichtung (12), die den kurzen Dateinamen aus weniger Zeichen durch Manipulieren des langen Dateinamens erzeugt.“

Im Streitpatent wird zu den neu hinzugekommenen Merkmalen ausgeführt, dass beim Anlegen einer Datei und der Vergabe eines langen Dateinamens immer auch ein kurzer 8.3-Dateiname automatisch erzeugt wird. Dabei werden die ersten sechs Buchstaben des langen Dateinamens genutzt. Darauf folgt eine Tilde und eine fortlaufende Nummer, die über die Anzahl der bereits mit diesen ersten sechs Buchstaben vergebenen 8.3-Dateinamen ermittelt wird (Streitpatentschrift, Absatz [0046]).

Die Beklagte führt in diesem Zusammenhang aus, dass es sich bei den neu hinzugekommenen Merkmalen um technische Merkmale handle, welche alle die Funktionsweise der Datenverarbeitungsanlage betreffen und deshalb bei der Beurteilung der erfinderischen Tätigkeit berücksichtigt werden müssten.

Nach Auffassung der Beklagten gebe die Druckschrift **D2** dem Fachmann keinerlei Anregung, einen kurzen Dateinamen i. S. d. Streitpatents zu erzeugen und diesen in einer Datenstruktur zu speichern. Vielmehr würde ein durch die Komponenten *filename*, *longfilename* und *ext* gebildeter langer Dateiname mit maximal 42 Zeichen für ältere Betriebssysteme auf die ersten 8 Zeichen des Dateinamens inklusive 3 Zeichen Extension abgekürzt, was möglicherweise zu identischen Dateinamen für verschiedene Dateien führe. Das Problem einer solchen Namenskollision werde in der Druckschrift **D2** erst gar nicht adressiert.

Dem Vorbringen der Beklagten kann nicht gefolgt werden. Bereits das Anlegen einer Datei und die Zuweisung eines langen Dateinamens zu dieser Datei durch einen Benutzer gehen nicht über die außertechnischen Vorgänge der Sammlung, Speicherung, Auswertung und Verwendung von Daten hinaus (BGH GRUR 2009, 479 – Steuerungseinrichtung für Untersuchungsmodalitäten; BGH a. a. O. – Webseitenanzeige). Als Maßnahmen der reinen Datenverarbeitung können die genannten Verfahrensschritte die Patentfähigkeit im Hinblick auf Art. 52 Abs. 2 Buchst. c oder d EPÜ jedenfalls nicht begründen.

Zwar lässt sich die in der Druckschrift **D2** im ersten Verzeichniseintrag der Struktur *struct \_baz* gespeicherte Zeichenkette mit nur 11 Zeichen aus dem langen Dateinamen durch eine bloße Namensverkürzung desselben ableiten, jedoch kann eine solche Art von „Manipulieren“ bzw. Verändern des langen Dateinamens die Forderung nach eindeutigen kurzen Dateinamen nicht notwendigerweise erfüllen.

Die im Patentanspruch 1 des Hilfsantrags 2 vorgeschlagene Maßnahme, aus langen Dateinamen heraus kurze Dateinamen mit weniger Zeichen zu erzeugen, trägt dem Wunsch nach Abwärtskompatibilität des Dateisystems Rechnung, in dem idealerweise lange Dateinamen umkehrbar eindeutig auf kurze Dateinamen abgebildet werden. Für eine solche Abbildungsvorschrift sind keinerlei auf technischen Überlegungen beruhenden Erkenntnisse erforderlich, sondern allenfalls Überlegungen aus der Programmierung und der mathematischen Logik, die einen

funktionellen Zusammenhang zwischen dem Aufbau des langen und des kurzen Dateinamens herstellen.

Da die neuen Merkmale des Patentanspruchs 1 gemäß Hilfsantrag 2 zu einer technischen Problemlösung nichts beitragen, ist auch der Hilfsantrag 2 nicht anders zu beurteilen als Hauptantrag und Hilfsantrag 1.

Somit kann dahingestellt bleiben, ob im Hilfsantrag 2 eine unzulässige Erweiterung vorliegt oder ob es sich bei dem neu hinzugekommenen Merkmal um ein klares Merkmal handelt.

**7.3** Das zusätzliche Merkmal des **Hilfsantrags 3** ist bei der Prüfung auf erfinderische Tätigkeit nicht zu berücksichtigen, weil es die Lösung eines technischen Problems mit technischen Mitteln weder bestimmt noch beeinflusst.

Gemäß Hilfsantrag 3 wird der Patentanspruch 1 nach Hilfsantrag 1 durch das folgende Merkmal eingeschränkt:

„wobei die Prüfsumme verwendet wird, um den zweiten Verzeichniseintrag mit dem ersten Verzeichniseintrag zu assoziieren.“

Entsprechend enthält der Nebenanspruch 11, der aus dem Patentanspruch 11 nach Hilfsantrag 1 hervorgeht, zusätzlich das Merkmal

„wobei die Prüfsumme verwendet wird, um den zweiten Verzeichniseintrag mit dem ersten Verzeichniseintrag zu assoziieren.“

In der Streitpatentschrift (Absätze [0031], [0041]) wird beschrieben, dass die Prüfsumme des kurzen Dateinamens dazu verwendet wird, den zweiten Verzeichniseintrag mit dem langen Dateinamen mit dem ersten Verzeichniseintrag, der den kurzen Dateinamen beinhaltet, zu verknüpfen (associate). Die Prüfsumme soll als Zeiger (pointer) auf den ersten Verzeichniseintrag dienen.

Die Beklagte argumentiert, dass der wesentliche Aspekt der Verwendung einer Prüfsumme darin bestehe, zwei Verzeichniseinträge mit langem und kurzem Dateinamen zu verknüpfen, und zwar über einen Dateinamen und nicht über eine Speicheradresse. Damit sei gewährleistet, dass auch nach einer Verschiebung des ersten Verzeichniseintrages (z. B. verursacht durch Programme zur Überprüfung von Dateisystem-Strukturen, sog. „disk utility programs“) innerhalb der FAT Partition im zweiten Verzeichniseintrag keine Aktualisierung eines Adresszeigers stattfinden müsse.

Das Merkmal der Verwendung einer Prüfsumme mit dem Ziel, zwei Verzeichniseinträge zu verbinden, löse demnach ein technisches Problem, da infolge der Verknüpfung Verzeichniseinträge lokalisiert werden könnten.

Der Senat stellt in diesem Zusammenhang fest, dass die genannte Prüfsumme bzw. „checksum“ nicht als Zeiger ausgelegt werden kann, dessen Bedeutung eine Speicheradresse ist. Weiterhin kann das Prüfsummenfeld mit 1 Byte Größe lediglich  $2^8 = 256$  verschiedene Werte annehmen, so dass nach Auffassung des Senats nicht mehr als 256 Dateien unterscheidbar wären und die Prüfsumme daher allenfalls als Verifizierung einer Verknüpfung, nicht aber als eindeutiger Verweis auf den kurzen Dateinamen dienen kann.

Diese Bedenken können aber zurücktreten. Falls mit der Prüfsumme eine Verknüpfung verbunden sein soll, löst sie auch kein technisches Problem mit technischen Mitteln. Vielmehr betrifft das neue Merkmal eine gedanklich logische Anweisung zur gegenseitigen Zuordnung von Verzeichniseinträgen, mithin eine Maßnahme zur Organisation von Daten, die allenfalls einen Bezug zur mathematischen Logik aufweist und weder einen neuen Aufbau einer Datenverarbeitungsanlage noch eine neue Art der Benutzung der Datenverarbeitungsanlage enthält. Daraus folgt auch, dass die von der Beklagten genannte Problemstellung, Verzeichniseinträge zu lokalisieren, das bloße Suchen und Auffinden von Information in Form von Datensätzen betrifft, was durch den logischen Vergleich von einzelnen Merkmalen bewerkstelligt wird. Das von der Beklagten formulierte Problem

liegt damit auf dem Gebiet der Logik, d. h. auf geisteswissenschaftlichem Gebiet, und ist nicht etwa einem Gebiet der Technik zuzurechnen.

Demnach ist der Gegenstand nach Patentanspruch 1 gemäß Hilfsantrag 3 durch die Druckschrift **D2** nahegelegt, da das neue Merkmal keinen Teilaspekt aufweist, der ein technisches Problem bewältigt.

Damit kann die Frage einer unzulässigen Erweiterung sowie fehlenden Ausführbarkeit dahinstehen.

**7.4 Die Hilfsanträge 4 und 5** können ebenfalls keinen Erfolg haben, weil auch die neu hinzugekommenen Merkmale nicht die Lösung eines technischen Problems mit technischen Mitteln bestimmen oder beeinflussen.

Gemäß Hilfsantrag 4 wird im Patentanspruch 1 nach Hilfsantrag 2 das Merkmal

„Erzeugen des kurzen Dateinamens aus weniger Zeichen durch Manipulieren des langen Dateinamens“

ersetzt durch

„Erzeugen des kurzen Dateinamens aus weniger Zeichen durch Kürzen des langen Dateinamens, so dass der kurze Dateiname eine Länge von acht Zeichen, ohne Erweiterung, nicht überschreitet.“

Entsprechend erhält der Nebenanspruch 11, der aus dem erteilten Patentanspruch 12 hervorgeht, anstelle des Merkmals aus Hilfsantrag 2

„(e) eine Einrichtung (12), die den kurzen Dateinamen aus weniger Zeichen durch Manipulieren des langen Dateinamens erzeugt.“

die zusätzliche Einschränkung

„(e) eine Einrichtung (12), die den kurzen Dateinamen aus weniger Zeichen durch Kürzen des langen Dateinamens erzeugt, so dass der kurze Dateiname eine Länge von acht Zeichen, ohne Erweiterung, nicht überschreitet.“

Gemäß Hilfsantrag 5 werden die in den Patentansprüchen 1 und 11 nach Hilfsantrag 4 hinzugefügten Merkmale noch dadurch weiter eingeschränkt, dass der erzeugte kurze Dateiname

„mit keinem anderen kurzen Dateinamen in Konflikt steht.“

Die in den Hilfsanträgen 4 und 5 neu aufgenommenen Merkmale dienen einer Konkretisierung der Lehre nach Hilfsantrag 2. Nach Vergabe eines langen Dateinamens soll der zugehörige kurze Dateiname dadurch gebildet werden, dass der kurze Dateiname aus einer Verkürzung des langen Dateinamens hervorgeht und maximal eine Länge von acht Zeichen (ohne Extension) aufweist (Hilfsantrag 4). In Hilfsantrag 5 wird noch präzisiert, dass der so gebildete kurze Dateiname nicht zu einem Konflikt mit anderen kurzen Dateinamen führen soll, also Namenskollisionen unter den kurzen Dateinamen ausgeschlossen werden sollen.

Im Streitpatent wird hierzu u. a. ausgeführt, dass der kurze Dateiname aus den ersten sechs Zeichen des langen Dateinamens zusammen mit einer Tilde sowie einer fortlaufenden Nummer gebildet wird (Streitpatentschrift, Absatz [0046]).

In Hinblick auf die obigen Ausführungen zum Hilfsantrag 2 können in den Präzisierungen der Hilfsanträge 4 und 5 allenfalls eine zusätzliche Bedingung für die Abbildung des langen auf den kurzen Dateinamen (Kürzen des langen Dateinamens und Beschränkung der Zeichen des kurzen Dateinamens) sowie der bloße Wunsch nach Eindeutigkeit bzw. Nichtverwechselbarkeit des kurzen Dateinamens erkannt werden.

In den neuen Merkmalen der Hilfsanträge 4 und 5 wird die Abbildungsvorschrift des langen auf den kurzen Dateinamen an die 8.3 Dateinamensbeschränkung

angepasst und die Forderung nach Eindeutigkeit für kurze Dateinamen formuliert, um für die Abwärtskompatibilität des Dateisystems zu sorgen. Die Merkmale beruhen sonach nicht auf technischen Überlegungen. Sie folgen lediglich den Regeln der mathematischen Logik und sind unter Anwendung herkömmlicher Programmiermethoden implementiert. Sie haben keinerlei Lösung eines technischen Problems mit technischen Mitteln zum Gegenstand und können bei der Beurteilung der erfinderischen Tätigkeit daher unberücksichtigt bleiben.

Die jeweiligen Lehren des Patentanspruchs 1 gemäß Hilfsantrag 4 und 5 sind nach allem aus den der Druckschrift **D2** entnehmbaren Stand der Technik nahegelegt und beruhen damit nicht auf erfinderischer Tätigkeit. Mit dem Patentanspruch 1 fällt jeweils der gesamte Hilfsantrag.

Die Frage, ob bei den Hilfsanträgen 4 und 5 eine unzulässige Erweiterung vorliegt, kann nach allem auch hier zurückstehen.

### III.

Die Kostenentscheidung beruht auf § 84 Abs. 2 PatG i. V. m. § 91 Abs. 1 Satz 1 ZPO. Die Entscheidung über die vorläufige Vollstreckbarkeit folgt aus § 99 Abs. 1 PatG, § 709 Satz 1 und 2 ZPO.

### IV.

Auf die anliegende Rechtsmittelbelehrung wird verwiesen.

Sredl

Merzbach

Baumgardt

Dr. Thum-Rung

Dr. Forkel

prä



## ***Rechtsmittelbelehrung***

Gegen dieses Urteil kann das Rechtsmittel der Berufung gemäß § 110 PatG eingelegt werden.

Die Berufung ist innerhalb eines Monats nach Zustellung des in vollständiger Form abgefassten Urteils - spätestens nach Ablauf von fünf Monaten nach Verkündung - durch einen in der Bundesrepublik Deutschland zugelassenen Rechtsanwalt oder Patentanwalt schriftlich zum Bundesgerichtshof, Herrenstraße 45a, 76133 Karlsruhe, einzulegen.

Die Berufungsschrift muss

- die Bezeichnung des Urteils, gegen das die Berufung gerichtet ist, sowie
- die Erklärung, dass gegen dieses Urteil Berufung eingelegt werde,

enthalten. Mit der Berufungsschrift soll eine Ausfertigung oder beglaubigte Abschrift des angefochtenen Urteils vorgelegt werden.

Auf die Möglichkeit, die Berufung nach § 125a PatG in Verbindung mit § 2 der Verordnung über den elektronischen Rechtsverkehr beim Bundesgerichtshof und Bundespatentgericht (BGH/BPatGerVV) auf elektronischem Weg zum Bundesgerichtshof einzulegen, wird hingewiesen (s. [www.bundesgerichtshof.de/erv.html](http://www.bundesgerichtshof.de/erv.html))