



BUNDESPATENTGERICHT

17 W (pat) 71/04

(Aktenzeichen)

Verkündet am
28. Juli 2011

...

BESCHLUSS

In der Beschwerdesache

betreffend die Patentanmeldung 102 32 674.6-53

...

hat der 17. Senat (Technischer Beschwerdesenat) des Bundespatentgerichts auf die mündliche Verhandlung vom 28. Juli 2011 unter Mitwirkung des Vorsitzenden Richters Dipl.-Phys. Dr. Fritsch, des Richters Dipl.-Ing. Prasch, der Richterin Dr. Mittenberger-Huber und des Richters Dipl.-Ing. Baumgardt

beschlossen:

Die Beschwerde wird zurückgewiesen.

Gründe

I.

Die vorliegende Patentanmeldung mit der Bezeichnung:

"Verfahren zur dynamischen Generierung strukturierter Dokumente"

ist am 18. Juli 2002 beim Deutschen Patent- und Markenamt unter Inanspruchnahme der Priorität der Voranmeldung DE 102 23 978 vom 29. Mai 2002 eingereicht worden.

Sie wurde von der Prüfungsstelle für Klasse G 06 F des Deutschen Patent- und Markenamts durch Beschluss vom 18. Mai 2004 mit der Begründung zurückgewiesen, dass der Gegenstand des Patentanspruchs 1 mangels erfinderischer Tätigkeit gegenüber dem ermittelten Stand der Technik nicht gewährbar sei.

Die hiergegen gerichtete Beschwerde der Anmelderin hat der erkennende Senat durch Beschluss vom 17. Januar 2008 mit der Begründung zurückgewiesen, dass das beanspruchte Verfahren nicht auf technischem Gebiet liege und daher keine patentfähige Erfindung i. S. d. § 1 Abs. 1 PatG sei.

Auf die zugelassene Rechtsbeschwerde hin hat der Bundesgerichtshof dem beanspruchten Verfahren sowohl technische Natur zugebilligt, als auch festgestellt, dass es nicht als Programm für Datenverarbeitungsanlagen nach § 1 Abs. 3 Nr. 3 i. V. m. Abs. 4 PatG vom Patentschutz ausgeschlossen ist (Beschluss Xa ZB 20/08 vom 22. April 2010 "Dynamische Dokumentengenerierung"). Die Sache wurde zur Prüfung der weiteren Anforderungen an die Patentfähigkeit einer Erfindung an das Patentgericht zurückverwiesen.

In der mündlichen Verhandlung vor dem Bundespatentgericht beantragt die Anmelderin und Beschwerdeführerin,

den Beschluss der Prüfungsstelle für Klasse G 06 F des Deutschen Patent- und Markenamts aufzuheben und das Patent 102 32 674 auf der Grundlage folgender Unterlagen zu erteilen:

Bezeichnung:

Verfahren zur dynamischen Generierung strukturierter Dokumente

Patentansprüche:

Patentansprüche 1 bis 7, vorgelegt mit Schriftsatz vom 15. März 2011,

hilfsweise

Patentansprüche 1 bis 6, vorgelegt mit Schriftsatz vom 15. März 2011,

Beschreibung:

Seiten 1, 8, 8a überreicht in der mündlichen Verhandlung vom 17. Januar 2008

Seiten 2 bis 7, 9 bis 13 vom Anmeldetag

Zeichnungen:

1 Blatt Zeichnungen mit zwei Figuren vom Anmeldetag.

Der geltende Patentanspruch 1 gemäß **Hauptantrag** lautet:

"Verfahren zur dynamischen Generierung strukturierter Dokumente (SD) an mindestens einem, mit einem Client (CL) kommunizierenden, in seinen Ressourcen limitierten, mikrocontroller-basierten Leitrechner (SRV), der als Kommunikationseinrichtung zur Vermittlung von Kommunikationsendgeräten ausgebildet ist, mit

- zumindest einem Vorlagedokument (TD) mit enthaltenen Aufrufen von Dienstnehmern (JB), das

§ sowohl in dem in seinen Ressourcen limitierten, mikrocontroller-basierten Leitreechner als auch in einem Leitreechner mit ausreichenden Ressourcen herangezogen werden kann,

§ für den Leitreechner ausreichender Ressourcen entwickelt und nach abgeschlossener Entwicklung in gleicher Weise auf dem in seinen Ressourcen limitierten, mikrocontroller-basierten Leitreechner herangezogen wird,

§ in Form einer Java Serverpage vorliegt, wobei die Dienstnehmer in Form von durch eine Java Virtual Machine bearbeitbaren JavaBeans vorliegen,

- einer Laufzeitumgebung eines Kontrollmoduls (CRT) zur Umsetzung des Vorlagedokuments, die dazu nicht mit einer Java Virtual Machine in Verbindung mit einer Servlet-Engine gebildet, sondern durch in der Sprache C++ programmierte Softwareimplementierungen im Kontrollmodul realisiert ist und im wesentlichen aus einem Parser und einem Scheduler besteht, wobei der Parser ohne eine Java Virtual Machine oder eine Servlet-Engine ausgestaltet ist,

- Leitreechnerkontrollfunktionen (SCF), die als architektur-, anwendungsprogramm- und/oder betriebssystemabhängige Befehle des Leitreechners ausgebildet sind,

- einem Schnittstellenmodul (IF) zum Zugriff auf die Leitreechnerkontrollfunktionen, bestehend aus einer anwendungsspezifischen, architekturabhängigen Software, die zumindest Datenbankzugriffe oder einen

Zugriff auf Konfigurationsparameter des Leitrechners mit einem definierten Befehlssatz gestattet,

das Verfahren umfassend die Schritte:

- a) Empfang von Anforderungsdaten (REQ) des Clients am Leitrechner,
- b) Extraktion von Anfrageparametern aus den Anforderungsdaten (REQ),
- c) dynamische Generierung des strukturierten Dokuments unter Verwendung mindestens eines der Vorlagedokumente,
 - i. wobei die Dienstnehmer unter Hinzuziehung der Anfrageparameter durch die Laufzeitumgebung aus dem Vorlagedokument extrahiert und auf einen korrespondierenden Befehlssatz des Schnittstellenmoduls abgebildet werden und nach derart in der Laufzeitumgebung erfolgter Ausführung Inhalte und/oder Struktur des strukturierten Dokuments definieren,
 - ii. wobei die Abbildung auf lediglich eine vordefinierte JavaBean-Klasse der JavaBean- bzw der Java Serverpage-Syntax beschränkt ist und Anfrageparameter ignoriert werden, für die kein zugehöriger Befehl im Befehlssatz des Schnittstellenmoduls existiert,
 - iii. wobei die Java Serverpage durch die Laufzeitumgebung nicht mit einer Java Virtual Machine in Verbindung mit einer Servlet-Engine korrekt umgesetzt wird, sondern indem die in der Java Serverpage enthaltenen JavaBeans bzw Aufrufe dieser JavaBeans aus der Java Serverpage durch den

Parser, der ohne eine Java Virtual Machine oder eine Servlet-Engine ausgestaltet ist, und den Scheduler extrahiert und auf den korrespondierenden Befehlssatz des Schnittstellenmoduls abgebildet werden,

- d) Übermittlung des dynamisch generierten strukturierten Dokuments an den Client."

Der geltende Patentanspruch 1 gemäß **Hilfsantrag** lautet:
(Änderungen gegenüber Hauptantrag kursiv).

"Verfahren zur dynamischen Generierung strukturierter Dokumente (SD) an mindestens einem, mit einem Client (CL) kommunizierenden, in seinen Ressourcen limitierten, mikrocontroller-basierten Leitreechner (SRV), der als Kommunikationseinrichtung zur Vermittlung von Kommunikationsendgeräten ausgebildet ist, mit

- zumindest einem Vorlagedokument (TD) mit enthaltenen Aufrufen von Dienstnehmern (JB), das

§ sowohl in dem in seinen Ressourcen limitierten, mikrocontroller-basierten Leitreechner als auch in einem Leitreechner mit ausreichenden Ressourcen herangezogen werden kann,

§ für den Leitreechner ausreichender Ressourcen entwickelt und nach abgeschlossener Entwicklung in gleicher Weise auf dem in seinen Ressourcen limitierten, mikrocontroller-basierten Leitreechner herangezogen wird,

§ in Form einer Java Serverpage vorliegt, wobei die Dienstnehmer in Form von durch eine Java Virtual Machine bearbeitbaren JavaBeans vorliegen,

§ auf einer beliebigen Rechneinheit, insbesondere der des Clients, gespeichert ist,

- einer Laufzeitumgebung eines Kontrollmoduls (CRT) zur Umsetzung des Vorlagedokuments, die dazu nicht mit einer Java Virtual Machine in Verbindung mit einer Servlet-Engine gebildet, sondern durch in der Sprache C++ programmierte Softwareimplementierungen im Kontrollmodul realisiert ist und im wesentlichen aus einem Parser und einem Scheduler besteht, wobei der Parser ohne eine Java Virtual Machine oder eine Servlet-Engine ausgestaltet ist,

- Leitrechnerkontrollfunktionen (SCF), die als architektur-, anwendungsprogramm- und/oder betriebssystemabhängige Befehle des Leitrechners ausgebildet sind,

- einem Schnittstellenmodul (IF) zum Zugriff auf die Leitrechnerkontrollfunktionen, bestehend aus einer anwendungsspezifischen, architekturabhängigen Software, die zumindest Datenbankzugriffe oder einen Zugriff auf Konfigurationsparameter des Leitrechners mit einem definierten Befehlssatz gestattet,

das Verfahren umfassend die Schritte:

- a) Empfang von Anforderungsdaten (REQ) des Clients am Leitrechner,
- b) Extraktion von Anfrageparametern aus den Anforderungsdaten (REQ),

- c) dynamische Generierung des strukturierten Dokuments unter Verwendung mindestens eines der Vorlagedokumente,
- i. *wobei von dem Kontrollmodul auf die Daten des Vorlagedokuments über ein paketorientiertes Netzwerk (NW) sowie über eine Ein-Ausgabeeinheit (IO) des Leitrechners zugegriffen wird,*
 - ii. wobei die Dienstnehmer unter Hinzuziehung der Anfrageparameter durch die Laufzeitumgebung aus dem Vorlagedokument extrahiert und auf einen korrespondierenden Befehlssatz des Schnittstellenmoduls abgebildet werden und nach derart in der Laufzeitumgebung erfolgter Ausführung Inhalte und/oder Struktur des strukturierten Dokuments definieren,
 - iii. wobei die Abbildung auf lediglich eine vordefinierte JavaBean-Klasse der JavaBean- bzw der Java Serverpage-Syntax beschränkt ist und Anfrageparameter ignoriert werden, für die kein zugehöriger Befehl im Befehlssatz des Schnittstellenmoduls existiert,
 - iv. wobei die Java Serverpage durch die Laufzeitumgebung nicht mit einer Java Virtual Machine in Verbindung mit einer Servlet-Engine korrekt umgesetzt wird, sondern indem die in der Java Serverpage enthaltenen JavaBeans bzw Aufrufe dieser JavaBeans aus der Java Serverpage durch den Parser, der ohne eine Java Virtual Machine oder eine Servlet-Engine ausgestaltet ist, und den Scheduler extrahiert und auf den korrespondierenden Befehlssatz des Schnittstellenmoduls abgebildet werden,

- d) Übermittlung des dynamisch generierten strukturierten Dokuments an den Client."

Den Ansprüchen 1 ist jeweils ein auf ein System gerichteter Anspruch 7 bzw. 6 nebengeordnet, der Mittel zur vollständigen Durchführung eines der beanspruchten Verfahren umfassen soll.

Zur Begründung ihrer Beschwerde führt die Anmelderin zunächst aus, dass der Bundesgerichtshof mit seiner Entscheidung die Hürde für die Anerkennung der Technizität von Software gesenkt habe. Für Software solle im Übrigen kein anderer Prüfungsstandard gelten als für andere Erfindungen. Dem Einwand, dass den Ansprüchen nur ein grundsätzliches Konzept und keine konkrete technische Ausgestaltung zu entnehmen sei, habe die Anmelderin mit einer Neufassung der Patentansprüche Rechnung getragen. Hinsichtlich der nunmehr zu erörternden Frage der erfinderischen Tätigkeit solle der technische Sachverhalt ganzheitlich bewertet werden. Dem Anmeldungsgegenstand liege das Problem zugrunde, ein Verfahren bereit zu stellen, das es ermöglicht, auch auf Vermittlungsrechnern mit beschränkten Ressourcen ein vom Client angefordertes strukturiertes Dokument dynamisch zu generieren. Gelöst werde dieses Problem dadurch, dass an Stelle einer Java Virtual Machine eine weniger rechen- und speicherplatzaufwändige spezielle Softwareimplementierung zum Einsatz komme.

Auch die nunmehr zum Stand der Technik genannte Druckschrift lege das beanspruchte Verfahren nicht nahe. Der Fachmann habe keinen Anlass gehabt, diese Druckschrift heranzuziehen. Dabei sei als Fachmann nicht jemand anzusetzen, der über breites Informatikwissen verfüge, sondern jemand, der auf dem Gebiet von Kommunikations- und Vermittlungsanlagen tätig sei. Das Verfahren und das System gemäß Hauptantrag beruhten daher auf erfinderischer Tätigkeit.

Das Verfahren nach Hilfsantrag unterscheide sich von dem gemäß Hauptantrag dadurch, dass die Vorlagedokumente auf einer beliebigen Rechneinheit im Netz gespeichert seien. Dies stelle einen Bruch mit der Denkwelt des Kommunikations-

fachmanns dar. Cloud-Computing, wie es derzeit propagiert werde, sei 2002 noch nicht bekannt gewesen.

II.

Die in rechter Frist und Form erhobene Beschwerde ist zulässig. Sie erweist sich jedoch nicht als begründet, da der Gegenstand der Anmeldung in den beantragten Fassungen nicht auf erfinderischer Tätigkeit beruht (§§ 1, 4 PatG).

1. In der Entscheidung über die Rechtsbeschwerde hat der Bundesgerichtshof festgestellt, dass der zum Patent angemeldete Gegenstand das Technizitätserfordernis des § 1 Abs. 1 PatG erfüllt, weil er die Nutzung von Komponenten einer Datenverarbeitungsanlage (Client, Leitrechner) lehrt. Weiterhin führt er aus, dass der Gegenstand der Anmeldung nicht als Programm für Datenverarbeitungsanlagen als solches i. S. v. § 1 Abs. 3 Nr. 3 i. V. m. Abs. 4 PatG anzusehen ist, da er verfahrensbestimmende Anweisungen enthält, die die Lösung eines konkreten technischen Problems mit technischen Mitteln zum Gegenstand haben.

2. Wie bereits in den vorhergehenden Beschlüssen erläutert, befasst sich die Anmeldung mit der dynamischen Generierung von Dokumenten in einer Client-/Server- oder Client-/Leitrechnerstruktur. Dabei wird davon ausgegangen, dass die Dokumente aus statischen (HTML-) Teilen und dynamischen (änderbaren) Teilen bestehen. Zur Anforderung eines solchen Dokuments übermittelt der Client Anforderungsdaten an den Leitrechner, d. h. eine Dokumentenadresse, bspw. URL, und Anfrageparameter, bspw. gewünschte Sprache. Die Dokumentenadresse benutzt der Leitrechner dazu, ein entsprechendes Vorlagedokument aus einem Speicher aufzurufen. Das Vorlagedokument besteht aus statischen Teilen und Dienstnehmern. Die Dienstnehmer werden entsprechend den Anfrageparametern von einer geeigneten Laufzeitumgebung (d.h. Softwareumgebung, vgl. S. 6, Z. 9 - 12 der Beschreibung) im Leitrechner ausgeführt und ergeben zusammen mit den stati-

schen Teilen das angeforderte Dokument. Das derart dynamisch generierte Dokument wird an den Client übermittelt.

Wie in der Beschreibung der Anmeldung dargestellt, kommt hierfür beim Leitrechner gewöhnlich eine standardisierte Laufzeitumgebung zum Einsatz, bspw. eine Java Virtual Machine in Verbindung mit einer Servlet-Engine (vgl. S. 4, Z. 25 - S. 5, Z. 12), die zwar extrem rechen- und speicherplatzaufwändig ist (vgl. den ergänzten Abs. auf S. 8, Z. 17 - 20), dafür aber großen Funktionsumfang aufweist. Ausgehend hiervon soll mit der Anmeldung ein Weg aufgezeigt werden, wie die Generierung strukturierter Dokumente mit dynamischem Inhalt auch auf Leitrechnern gelingen kann, die in ihren Ressourcen begrenzt sind und auf denen daher keine anspruchsvolle Laufzeitumgebung installiert werden kann. Dabei soll eine Portierung der Vorlagendokumente zwischen Leitrechnern mit ausreichenden Ressourcen und ressourcenbegrenzten Leitrechnern möglich sein (vgl. S. 7, Z. 5 - 10), d. h. die für eine anspruchsvolle Laufzeitumgebung entworfenen Dokumente sollen auch auf einem ressourcenbegrenzten Leitrechner generiert werden können.

3. Zur Lösung dieser Problemstellung schlägt Patentanspruch 1 gemäß Hauptantrag vor, an Stelle einer standardisierten aber ressourcenaufwändigen Laufzeitumgebung wie einer Java Virtual Machine eine spezielle Laufzeitumgebung in Form einer "schlankeren Softwareimplementierung" (vgl. S. 8, Z. 17 - 22) vorzusehen, die im Wesentlichen aus einem Parser und einem Scheduler besteht und in der Sprache C++ implementiert ist (vgl. Anspruch 1, zweiter Spiegelstrich und Merkmal iii.).

Darüber hinaus schlägt Merkmal ii. vor, nicht alle Dienstnehmer der JavaServerpage-Syntax abzubilden, sondern nur eine (nicht näher) vordefinierte JavaBean-Klasse, und einen (nicht näher bestimmten) Teil der Anfrageparameter zu ignorieren.

Es ist nachvollziehbar, dass die dynamische Generierung von Dokumenten unter Verwendung einer "schlankeren" Laufzeitumgebung und der Abbildung nur eines

Teils der Dienstnehmer und Anfrageparameter weniger Ressourcen erfordert als der Einsatz einer standardisierten Laufzeitumgebung unter Abbildung aller Dienstnehmer und Parameter, wie diese eine Java Virtual Machine leistet.

Insofern ist der Auffassung der Anmelderin beizutreten, dass im vorliegenden Fall alle Merkmale des Anspruchs 1 zur Lösung der erläuterten technischen Problemstellung beitragen und daher auch bei der Prüfung von Neuheit und erfinderischer Tätigkeit zu berücksichtigen sind.

4. Das Verfahren nach Patentanspruch 1 gemäß Hauptantrag beruht nicht auf erfinderischer Tätigkeit.

Als Stand der Technik ist hierbei folgende Druckschrift von besonderer Bedeutung:

Aufsatz von Michael O'Brien: "Embedded Web Servers", abgedruckt im US-Magazin "Embedded Systems Programming" November 1999, S. 67 - 72.

Dieser Aufsatz befasst sich mit "Embedded Web Servers", d. h. mit der Gestaltung von Software, mit der eine dynamische Generierung von Webseiten (dynamic page generation) auf "embedded Servern" möglich sein soll (vgl. S. 68 mittl. Spalte). "Embedded Server" sind (Leit-) Rechner, die in einen technischen Kontext eingebettet sind und bspw. Heizungsanlagen steuern. Damit solche Server von einem Client (remote computer) über das Web bzw. Internet überwacht und gesteuert werden können, sind sie mit einem "piece of software called an embedded web server" ausgestattet (vgl. S. 67 Zusammenfassung). Wie auf S. 67, rechte Sp. bis S. 68 li. Sp. erläutert, sind "embedded Server" kostenlimitiert und verfügen nur über eingeschränkte Speicher- und CPU-Ressourcen, die den Einsatz standardisierter Software für die Generierung von Webseiten im Server bzw. Leitrechner nicht zulassen. Insoweit befasst sich der Aufsatz von Michael O'Brien mit der gleichen Problemstellung wie die Anmeldung.

Dem hiergegen gerichteten Argument der Anmelderin, zuständiger Fachmann sei ein auf dem Gebiet der Kommunikations- oder Vermittlungstechnik tätiger Ingenieur, der nicht über ein breites Informatikwissen verfüge und den Inhalt dieses Aufsatzes nicht herangezogen hätte, vermochte der Senat nicht zu folgen. Denn Aufsatz wie beanspruchtes Verfahren befassen sich mit den Vorgängen in einem Client-Serversystem unter Verwendung eines Rechnernetzwerks (Web) und nicht mit Vermittlungstechnik, wie sie bei Kommunikationsendgeräten wie Telefonen verwendet wurde. Im Detail befasst sich das beanspruchte Verfahren mit der dynamischen Generierung von strukturierten Dokumenten per Webserverprogrammen, so dass der gleichfalls auf die Konzeption von Webserverprogrammen und die Generierung von Webseiten gerichtete Aufsatz durchaus als relevanter Stand der Technik anzusehen ist. Als Fachmann ist folglich ein Informatiker oder Softwareingenieur anzusetzen, der über praktische Erfahrung in der Programmierung von vernetzten Client-/Server-Systemen verfügt und jedenfalls mit den Grundlagen des Webs bzw. des Internets vertraut ist.

Dieser Fachmann entnimmt dem Aufsatz von Michael O'Brien zunächst die grundsätzlichen Abläufe bei der dynamischen Generierung von Dokumenten (dynamic page generation, on the fly) mit Absetzen einer Anforderung (durch Übersenden von Anforderungsdaten) des Clients (remote computer) an den Server (embedded computer) und Übermittlung des dort dynamisch generierten Dokuments an den Client, wie sie in den Merkmalen a) und d) des Anspruchs 1 zum Ausdruck kommt. Dabei wird in Übereinstimmung mit Anspruch 1 davon ausgegangen, dass in einem Speicher (memory) eine Anzahl von Dokumenten (pages) gespeichert ist, deren Inhalt mindestens teilweise dynamisch erzeugt wird, d. h. die Dienstnehmer enthalten (vgl. S. 67 linke Sp., S. 68 mittl. Sp., Abs. 2).

In Übereinstimmung mit der dem Anmeldungsgegenstand zugrunde liegenden Problemstellung wird davon ausgegangen, dass bei "embedded" Servern die Software von standardisierten Web-Servern nicht ablauffähig ist, weil sie für hoch leistungsfähige Server (-Hardware) entworfen wurde. Die Entwerfer von Serversoft-

ware für eingebettete Systeme müssten berücksichtigen, dass wenig Speicherbedarf und Verarbeitungskapazität (memory footprint, resource expectations) zur Verfügung stehen. Dabei soll aber eine dynamische Generierung von strukturierten Dokumenten (dynamic page generation) weiter möglich sein (vgl. S. 67, re. Sp. - S. 68 mittlere Spalte).

Zur Lösung dieser Problematik wird im Abschnitt "Embedded JavaScript" des Aufsatzes (ab S. 70 Mitte) vorgeschlagen, eine Laufzeitumgebung zu schaffen, in der nur ein "subset" der JavaScript-Befehle implementiert ist, die also nur einen vordefinierten Ausschnitt des vollständigen Befehlsvorrats interpretieren kann. Hinsichtlich der Auswahl der Befehle, die zu implementieren bzw. wegzulassen sind, wird beispielhaft auf den "GoAhead Webserver" verwiesen. Aus der Anweisung, dass nur ein Ausschnitt der Befehle implementiert werden soll, schließt der Fachmann, dass in diesem Zusammenhang auch die Anfrageparameter, für die kein zugehöriger Befehl (mehr) existiert, zu ignorieren sind, wie Merkmal ii.) des Anspruchs 1 angibt. Durch die Implementierung nur eines Ausschnitts der Befehle in dem "embedded WebServer" soll sich der Speicherbedarf von 200K+ auf 15K reduzieren.

Der Fachmann entnimmt dem Aufsatz von Michael O'Brien sonach, dass die dynamische Generierung strukturierter Dokumente auch auf Servern mit eingeschränkten Ressourcen möglich ist, wenn anstelle einer aufwändigen standardisierten Laufzeitumgebung, bspw. eines vollständigen JavaScript-Interpreters, eine Laufzeitumgebung vorgesehen wird, die speziell auf die begrenzten Ressourcen des Leitrechners zugeschnitten ist und nur einen Ausschnitt des Befehlsvorrats interpretiert. Damit entnimmt der Fachmann dem Aufsatz von Michael O'Brien den entscheidenden Hinweis, wie er vorzugehen hat, um strukturierte Dokumente auch auf Servern bzw. Leitrechnern generieren zu können, die in ihren Ressourcen so begrenzt sind, dass auf ihnen keine standardisierte Laufzeitumgebung ablauffähig ist. Es liegt keine erfinderische Tätigkeit darin, diese Lehre auf eine Java Virtual Machine und dafür vorgesehene Dienstnehmer (JavaBeans) zu übertragen.

Die Anmelderin wendet hiergegen ein, dass der Aufsatz von Michael O'Brien lediglich vorschläge, die standardisierte Laufzeitumgebung des Webservers, bspw. die Java Virtual Machine "abzustrippen", während die Anmeldung eine grundlegend neue (und schlankere) Implementierung der Laufzeitumgebung mit Parser und Scheduler vorschläge.

Der Auffassung, dass der Aufsatz nur ein Weglassen von Teilen der die Befehle interpretierenden Software vorschläge, kann nicht gefolgt werden. Auf Seite S. 70, re. Sp., 2. Abs. ist ausgeführt: "such an implementation of a JavaScript subset can be as small as an 15K embeddable JavaScript Interpreter". Der Fachmann versteht diesen Satz in der Weise, dass von den Befehlen, die in der Scriptsprache JavaScript definiert sind und die für die Erstellung von Dokumenten verwendet werden können, ein Teil weggelassen wird, nicht aber, dass der JavaScript Interpreter, also die Laufzeitumgebung, von der das Subset von Befehlen interpretiert bzw. ausgeführt wird, durch Weglassen von Teilen erzeugt wird. Für diese Auslegung spricht auch, dass in Übereinstimmung mit Merkmal iii.) zur Interpretation der Dokumente ebenfalls eine (Neu-) Implementierung mit einem Parser vorgeschlagen wird, wodurch sich minimale Anforderungen an die Rechenleistung (minimal CPU requirements) des Servers ergeben sollen (vgl. S. 70, re. Sp., 3. Abs.). Würden nur Teile der standardisierten Laufzeitumgebung wie der Java Virtual Machine weggelassen, so würde sich die (Neu-) Implementierung eines Parsers erübrigen.

Eine erfinderische Besonderheit kann auch in den weiter im Anspruch 1 genannten Merkmalen und Präzisierungen nicht erkannt werden.

In dem Umstand, dass das Vorlagedokument in Form einer Java Serverpage vorliegt und entsprechend JavaBeans als Dienstnehmer zur Anwendung kommen, erkennt der Fachmann nur die Festlegung auf einen der Standards, die auch in der Beschreibung einleitend als bekannt erläutert werden. In Hinsicht auf die mit dem Aufsatz von Michael O'Brien vorgeschlagene Vorgehensweise bei der Generierung von dynamischen Dokumenten bei ressourcenbegrenzten Leitrechnern hat diese Festlegung keinen wesentlichen Einfluss.

Eine erfinderische Leistung kann auch nicht darin erkannt werden, dass die Programmierung der (speziellen) Laufzeitumgebung in der Programmiersprache C++ erfolgen soll. Dabei handelt es sich um eine gängige Programmiersprache. Die Anmelderin hat auch nicht dargetan, dass die Verwendung dieser Programmiersprache in technischer Hinsicht zu vorteilhaften Lösungen führt.

Ebenso wenig kann in der Verwendung eines Schedulers neben dem Parser (Merkmal iii.) eine erfinderische technische Ausprägung erkannt werden. Beide Softwaremodule sind dem Fachmann als Bestandteile von Laufzeitumgebungen für die Ausführung von Programmen geläufig. Während Parser der Analyse der Struktur von Programmen dienen, sorgen Scheduler dafür, dass diese Struktur in der richtigen Reihenfolge ausgeführt wird.

Schließlich kann auch in der in Anspruch 1 angegebenen Gliederung der Software des Leitrechners in betriebssystemabhängige Leitrechnerkontrollfunktionen, anwendungsspezifisches Schnittstellenmodul und Laufzeitumgebung bzw. Kontrollmodul keine besondere technische Leistung erkannt werden. Dass die Software von Servern bzw. Leitrechnern vorteilhaft in Module gegliedert ist, ergibt sich für den Informatiker schon aus den unterschiedlichen auszuführenden Funktionen und aus Gründen der besseren Übersichtlichkeit und Wartbarkeit. Im Übrigen ist nicht ersichtlich, in welcher Weise gerade die angegebene Gliederung in Hinsicht auf die Generierung strukturierter Dokumente Vorteile aufweist. Solche wurden auch von der Anmelderin nicht geltend gemacht.

Das Verfahren zur dynamischen Generierung strukturierter Dokumente nach dem Patentanspruch 1 ist dem Fachmann folglich durch die Ausführungen in dem Aufsatz von Michael O'Brien nahe gelegt.

5. Dem Anspruch 1 ist der auf ein System zur dynamischen Generierung strukturierter Dokumente gerichtete Anspruch 7 nebengeordnet. Er soll "Mittel zur vollständigen Durchführung" des Verfahrens nach Anspruch 1 umfassen, ohne dass

diese konkret dargelegt werden. Nachdem das Verfahren nach Anspruch 1 nicht patentfähig ist, kann folglich auch Anspruch 7 keinen Bestand haben.

Dem Hauptantrag der Anmelderin war daher nicht zu folgen.

6. Der Patentanspruch 1 gemäß Hilfsantrag unterscheidet sich vom Hauptantrag dadurch, dass die Vorlagendokumente "auf einer beliebigen Rechneinheit" gespeichert sein können, auf die gemäß Merkmal i. über ein paketorientiertes Netzwerk sowie über eine Ein-Ausgabeeinheit des Leitrechners zugegriffen wird.

In dem Umstand, dass die Vorlagen für die dynamische Generierung von Dokumenten auf einer beliebigen Rechneinheit des Netzwerkes gespeichert sein und vom Leitrechner abgerufen werden können, kann keine erfinderische Besonderheit erkannt werden. Denn zu dem Grundlagenwissen über das Web bzw. Internet gehörte bereits zum Prioritätszeitpunkt, dass dort ein globales (URL-) Adressierschema verwendet wurde, das es ermöglichte, im Netz jederzeit auf alle Informationen zuzugreifen, unabhängig von dem physikalischen Aufenthaltsort des Rechners bzw. Clients auf dem sie gespeichert sind. Diesen Sachverhalt entnimmt der Fachmann im Übrigen auch dem Abs. 1 des Kapitels "URL-Handler" auf S. 72 des Aufsatzes von Michael O'Brien.

Dem Grundwissen des Fachmanns ist ebenso zuzurechnen, dass es sich bei dem Web bzw. Internet um ein paketorientiertes Netzwerk handelt, auf das Rechner üblicherweise über eine Ein-Ausgabeeinheit zugreifen.

Das Verfahren nach Anspruch 1 gemäß Hilfsantrag beruht daher ebenfalls nicht auf erfinderischer Tätigkeit. Sonach konnte auch dem Hilfsantrag der Anmelderin nicht gefolgt werden. Die Beschwerde war daher erneut zurückzuweisen.

7. Die Anmelderin hat angeregt, hinsichtlich der Frage der erfinderischen Tätigkeit bei Erfindungen auf dem Gebiet der Software Rechtsbeschwerde zuzulassen.

Die Rechtsbeschwerde ist zuzulassen, wenn eine Rechtsfrage grundsätzlicher Bedeutung zu entscheiden ist oder die Fortbildung des Rechts oder die Sicherung einer einheitlichen Rechtsprechung eine Entscheidung des Bundesgerichtshofs erfordert (§ 100 Abs. 2 PatG).

Eine uneinheitliche Rechtsprechung der Senate des Bundespatentgerichts in Hinsicht auf die Frage der erfinderischen Tätigkeit bei Erfindungen mit Bezug zur Datenverarbeitung hat die Anmelderin nicht geltend gemacht; eine solche ist auch für den Senat nicht ersichtlich. Hinsichtlich der Frage, welcher Sachverhalt bei der Bewertung der erfinderischen Tätigkeit bei solchen Erfindungen zugrunde liegen soll, geht der Senat, wie offenbar auch die Anmelderin, im vorliegenden Fall davon aus, dass alle Merkmale zu berücksichtigen sind.

Auch eine Rechtsfrage grundsätzlicher Bedeutung liegt nicht mehr vor. Denn der Bundesgerichtshof hat zu dieser Frage in der vorliegenden Sache bereits in der Entscheidung über die Rechtsbeschwerde grundsätzlich Stellung bezogen. In Abs. [0023] der Entscheidung ist ausgeführt, dass bei der Prüfung von Neuheit und erfinderischer Tätigkeit die Lösung des technischen Problems in den Blick zu nehmen ist. Diese Auffassung wurde vom Bundesgerichtshof in dem Urteil X ZR 47/07 vom 26. Oktober 2010 bekräftigt (vgl. Leitsatz b und Abs. [0031] "Wiedergabe topografischer Informationen").

Der Anregung der Anmelderin auf Zulassung der Rechtsbeschwerde war daher nicht zu folgen.

Dr. Fritsch

Prasch

Dr. Mittenberger-Huber

Baumgardt

Fa