

**Interner Verteilerschlüssel:**

- (A) [ - ] Veröffentlichung im ABl.
- (B) [ - ] An Vorsitzende und Mitglieder
- (C) [ - ] An Vorsitzende
- (D) [ X ] Keine Verteilung

**Datenblatt zur Entscheidung  
vom 2. Juli 2014**

**Beschwerde-Aktenzeichen:** T 2270/10 - 3.5.06

**Anmeldenummer:** 07012983.8

**Veröffentlichungsnummer:** 2012227

**IPC:** G06F9/44

**Verfahrenssprache:** DE

**Bezeichnung der Erfindung:**

Programmieroberfläche zum Programmieren von Computern

**Anmelder:**

Renner, Peter

**Stichwort:**

Programmieroberfläche/RENNER

**Relevante Rechtsnormen:**

EPÜ 1973 Art. 56

VOBK Art. 15(3), 15(5), 15(6)

**Schlagwort:**

Mündliche Verhandlung - Fernbleiben von der mündlichen  
Verhandlung

Erfinderische Tätigkeit - (nein)

**Zitierte Entscheidungen:**

G 0003/08, T 0641/00, T 1171/06, T 1539/09

**Orientierungssatz:**



**Beschwerdekammern  
Boards of Appeal  
Chambres de recours**

European Patent Office  
D-80298 MUNICH  
GERMANY  
Tel. +49 (0) 89 2399-0  
Fax +49 (0) 89 2399-4465

Beschwerde-Aktenzeichen: T 2270/10 - 3.5.06

**E N T S C H E I D U N G**  
**der Technischen Beschwerdekammer 3.5.06**  
**vom 2. Juli 2014**

**Beschwerdeführer:** Renner, Peter  
(Anmelder) Heiligenstock 2  
51515 Kürten (DE)

**Vertreter:** Lippert, Stachow & Partner  
Patentanwälte  
Postfach 30 02 08  
51412 Bergisch Gladbach (DE)

**Angefochtene Entscheidung:** Entscheidung der Prüfungsabteilung des Europäischen Patentamts, die am 24. Juni 2010 zur Post gegeben wurde und mit der die europäische Patentanmeldung Nr. 07012983.8 aufgrund des Artikels 97 (2) EPÜ zurückgewiesen worden ist.

**Zusammensetzung der Kammer:**

**Vorsitzender** D. Rees  
**Mitglieder:** M. Müller  
W. Sekretaruk

## Sachverhalt und Anträge

I. Die Prüfungsabteilung wies die Europäische Patentanmeldung Nr. 07012983.8 mit der Begründung zurück, die beanspruchte Erfindung weise gegenüber dem Dokument

D1: EP 1 610 219 A1

keine erfinderische Tätigkeit auf, Artikel 56 EPÜ.

II. Gegen diese Entscheidung legte die Anmelderin am 29. Juli 2010 Beschwerde ein. Die Beschwerdegebühr wurde am 28. Juli 2010 entrichtet und die Beschwerdebegründung ging am 4. November 2010 ein.

III. Die Beschwerdeführerin beantragt, den Beschluss der Prüfungsabteilung aufzuheben und ein Patent zu erteilen auf der Basis geänderter Ansprüche 1-30 gemäß einem Hauptantrag und Hilfsanträgen 2 und 3, Ansprüchen 1-29 gemäß einem Hilfsantrag 1 oder Ansprüchen 1-28 gemäß einem Hilfsantrag 4, die mit der Beschwerdebegründung eingereicht wurden. Die übrigen Anmeldungsunterlagen sind wie folgt:

Zeichnungen, Blatt

1-8 eingegangen mit Schreiben vom 13. Nov. 2007

Beschreibung, Seiten

*Hauptantrag und Hilfsantrag 1 oder 2*

1, 3, 3a, 27-29, 31, 33, 35, 36, 38-43

eingegangen mit der Beschwerdebegründung

2, 4-26, 30, 32, 34, 37

wie ursprünglich eingereicht

*Hilfsantrag 3 oder 4*

1, 3, 8, 14, 15, 17-19, 21-29, 31, 33, 35, 36, 38-43

eingegangen mit der Beschwerdebegründung

2, 4-7, 9-13, 16, 20, 30, 32, 34, 37

wie ursprünglich eingereicht

- IV. Mit einer Ladung zur mündlichen Verhandlung teilte die Kammer der Beschwerdeführerin ihre vorläufige Meinung mit, dergemäß der beanspruchte Gegenstand schon deshalb keine erfinderische Tätigkeit aufwiese, weil er gegenüber dem allgemeinen Fachwissen keine technische Aufgabe lösen würde, Artikel 56 EPÜ 1973. Darüber hinaus wiese der beanspruchte Gegenstand keine erfinderische Tätigkeit gegenüber D1 auf, da die Unterschiedeigenschaften entweder fachüblich seien oder kein erkennbares technisches Problem lösten. Schließlich wiesen Ansprüche aller Anträge Klarheitsmängel auf, Artikel 84 EPÜ 1973.
- V. Auf die Ladung ging keine Erwiderung der Beschwerdeführerin ein.
- VI. Die mündliche Verhandlung fand am 2. Juli 2014 in Abwesenheit der Beschwerdeführerin statt.
- VII. Anspruch 1 des Hauptantrags lautet wie folgt:

"Computergestütztes Verfahren zur imperativen Programmierung von Computern mit mehreren Programmanweisungen umfassendem Programmcode, wobei Schlüsselworte und Auswahllisten vorgesehen sind, mit einer grafischen Benutzeroberfläche wobei sich Programmanweisungen aus Operationen und Objekten zusammensetzen, wobei Objekte adressierbar sind sowie Operationen und Objekte aus Auswahllisten ausgewählt werden,

dadurch gekennzeichnet, dass

einem individuellen Namen eines Objektes, das durch das Ersetzen eines Schlüsselwortes in eine Anweisung

eingefügt wird, ein Pfad zugeordnet ist, welcher zusammen mit dem individuellen Namen in die Anweisung eingefügt wird, wenn dieses Objekt in einer hierarchischen Struktur eingebunden ist."

Anspruch 1 des 1. Hilfsantrags entspricht Anspruch 1 des Hauptantrags, an dessen Ende der folgende Text angefügt wurde:

"... und dass ein Anweisungsabschnitt, mit dem ein Schlüsselwort in der Anweisung ersetzt wird und der den individuellen Namen eines Objektes sowie den möglicherweise zugeordneten Pfad umfasst, eine markierte Stelle aufweist, mit der die Auswahlliste wieder aufgerufen werden kann, mit welcher der Anweisungsabschnitt erstellt worden ist."

Anspruch 1 des 2. und 3. Hilfsantrags stimmen mit Anspruch 1 des Hauptantrags in der Präambel überein, während der kennzeichnende Teil jeweils ersetzt wurde. Der kennzeichnende Teil von Anspruch 1 des 2. Hilfsantrags lautet wie folgt:

"... dadurch gekennzeichnet, dass nach dem Ersetzen eines Schlüsselwortes durch einen Anweisungsabschnitt in diesem Abschnitt eine markierte Stelle vorgesehen ist, mit der die Auswahlliste wieder aufgerufen werden kann, mit welcher der Anweisungsabschnitt erstellt worden ist."

Der kennzeichnende Teil von Anspruch 1 des 3. Hilfsantrags lautet wie folgt:

"... dadurch gekennzeichnet, dass Objekte Kategorien zugeordnet sind, wobei als Kategorien Hauptobjekte Objekte und Subobjekte vorgesehen sind, wobei Hauptobjek-

te weitere Hauptobjekte, Objekte oder Subobjekte und Datensätze (Datenbanken, Dateien) enthalten können, wobei Objekte und Subobjekte Daten enthalten, wobei Hauptobjekte, Objekte und Subobjekte in den Auswahllisten in verschiedenen Ebenen zugeordnet sind, und wobei Subobjekte entweder einen Bezug zu einem Hauptobjekt, zu einem Objekt oder zu einer Operation haben."

Anspruch 1 des 4. Hilfsantrags stimmt mit Anspruch 1 des 1. Hilfsantrags überein, wobei zwischen die Wörter "dadurch gekennzeichnet, dass" und die Merkmale des kennzeichnenden Teils von Anspruch 1 des 1. Hilfsantrags die Merkmale des kennzeichnenden Teils des 3. Hilfsantrags eingefügt wurden.

Alle Anträge enthalten zusätzlich einen unabhängigen Anspruch auf ein computergestütztes System, der weitgehend den entsprechenden Verfahrensansprüchen entspricht, nämlich Anspruch 23 des Hauptantrags, Anspruch 19 des 1. Hilfsantrags, Anspruch 20 des 2. und 3. Hilfsantrags und Anspruch 18 des 4. Hilfsantrags. Der genaue Wortlaut der Systemsansprüche ist für die Entscheidung der Kammer nicht von Bedeutung.

VIII. Am Ende der mündlichen Verhandlung verkündete der Vorsitzende die Entscheidung der Kammer.

### **Entscheidungsgründe**

1. Die ordnungsgemäß geladene Beschwerdeführerin war in der mündlichen Verhandlung nicht anwesend. Gemäß Artikel 15 (3) VOBK wurde das schriftliche Vorbringen der Beschwerdeführerin berücksichtigt. Am Ende der mündlichen Verhandlung verkündete die Kammer ihre Entschei-

dung, da die Sache entscheidungsreif war (Artikel 15 (5,6) VOBK) und die Abwesenheit der Beschwerdeführerin kein Grund war, die Entscheidung aufzuschieben (Artikel 15 (3) VOBK).

2. Die folgenden Entscheidungsgründe beschränken sich im Wesentlichen auf die vorläufige Meinung der Kammer, die im Ladungszusatz der Beschwerdeführerin mitgeteilt worden ist.

### *Die Erfindung*

3. Die Anmeldung geht von der Beobachtung aus, dass die konventionelle Programmierung aufgrund ihrer Komplexität nur noch Spezialisten offenstehe und befasst sich mit der Aufgabe, diesen Umstand zu ändern (ursprüngliche Anmeldung, S. 2, 1. Satz; S. 3, 1. Abs.).
  - 3.1 Als Lösung schlägt die Anmeldung ein Programmiersystem und das entsprechende Verfahren vor, das den Programmierer "vom Ballast der Sprachkonventionen und dem komplizierten Formalismus ... befreien" soll (S. 3, 1. Satz). Dieses Ziel wird erfindungsgemäß durch eine grafische Benutzeroberfläche erzielt, die Programmkomponenten zur Auswahl aus Listen (Kontextmenüs) bereitstellt und so hilft, das mühsame und fehleranfällige Eintippen von Text zu vermeiden (S. 3, letzter Abs.).
  - 3.2 Gleichzeitig sind Programmanweisungen vorgesehen, die grundsätzlich aus Operationen und Objekten bestehen (S. 3, vorletzter Abs.), so dass der Programmierer sich auf den zu steuernden Prozess - das "Prozessuale" - konzentrieren kann (S. 4, 1. Abs.). Auf diese Weise werde das Programmieren intuitiver und der Programmcode selbst leichter zu lesen und zu warten (S. 4, 1. Abs.).

- 3.3 Die Anmeldung stellt fest (S. 5, 2. Abs.), dass "Objekte ... häufig in Baumstrukturen eingebunden [sind], so dass diese mit einem Pfad gekoppelt sein können". Damit laute die typische Grundform einer Anweisung wie folgt: ">Operation: Pfad\Objekt<". Die Teile "Operation" und "Pfad\Objekt" nehmen hierin beschreibungsgemäß die "Funktion von Schlüsselworten" ein, die im "Bedienprozess" bei der Erstellung einer konkreten Anweisung durch "eine spezielle Operation und Objekte ersetzt werden können" (S. 5, letzter Abs.). So erstellte Anweisungen können in dem Abschnitt, in dem eine Operation oder ein Objekt nach Auswahl aus einer Liste eingefügt wurde, eine markierte Stelle aufweisen, mittels derer die relevante Auswahlliste wieder aufgerufen werden kann (S. 28, 2. Abs. und ursprünglicher Anspruch 18). Es wird beschrieben, dass eine solche Markierung vom Softwareentwickler angeklickt werden könne, um so etwa ein gewähltes Objekt durch ein anderes zu ersetzen.
- 3.4 Es sind unterschiedliche Kategorien von Objekten vorgesehen, die als "Hauptobjekte, Objekte und Subobjekte" bezeichnet werden (S. 8, 2.-4. Abs.). Die Beschreibung erläutert, dass "Hauptobjekte" Objekte enthalten und somit zum Beispiel die "Funktion von Containern" annehmen können, während "Objekte" keine weiteren Objekte enthalten. Sie erläutert weiter, dass "Subobjekte" einen "Bezug zu einem Objekt oder zu einer Operation haben" und so "die korrespondierende Operation oder das Objekt ergänzen". "Welche Hauptobjekte, Objekte und Subobjekte einem Softwareentwickler zur Verfügung stünden, hinge "von der Mächtigkeit der Sprache ab, die dem Softwareentwickler zur Verfügung steht" und "davon, welche Objekte er installiert hat" (S. 9, Zn. 1.-4.).

### *Technischer Charakter*

4. Die Erfindung gemäß Anspruch 1 aller Anträge bezieht sich auf ein "[c]omputergestütztes Verfahren zur imperativen Programmierung" unter Verwendung einer "grafischen Benutzeroberfläche" sowie das entsprechende System. Allein schon wegen der Verwendung eines Computers handelt es sich dabei nach gefestigter Rechtsprechung der Beschwerdekammern (vgl. etwa G 3/08, ABl. EPA, 2011, 10; Gründe 10.7) um Gegenstände, die technischen Charakter haben und somit unter Artikel 52 (2,3) EPÜ nicht zu bemängeln sind.

### *Auslegung der Ansprüche*

5. Einige Merkmale der beanspruchten Erfindung definieren Aspekte einer Programmiersprache, andere Merkmale definieren Aspekte der zugehörigen Programmierumgebung, insbesondere ihrer grafischen Benutzeroberfläche.
  - 5.1 Die vorgeschlagene Programmiersprache ist gemäß Anspruch 1 des Hauptantrags dadurch charakterisiert, dass sie "imperativ" ist, dass sich ihre Anweisungen aus "Operationen" und "Objekten" zusammensetzen, dass Objekte in einer "hierarchischen Struktur eingebunden" sein können und durch einen "Namen" und ggf. einen "Pfad" adressierbar sind. Zur Definition der Programmiersprache gehört nach Ansicht der Kammer weiterhin, dass man zulässige Anweisungen durch das Ersetzen von "Schlüsselwörtern" durch Operationen und Objekte aus vorgegebenen "Listen" erhält, insofern als die Syntax der Programmiersprache festlegt, welche Schlüsselwörter an welchen Positionen innerhalb von Anweisungen zulässig sind und durch welche Namen sie im Einzelnen ersetzt werden dürfen. Keines dieser Merkmale ist nach Ansicht der Kammer der grafischen Benutzeroberfläche

- zuzuordnen. Nur die Formulierungen, dass Operationen und Objekt aus "Auswahl"listen "ausgewählt" und entsprechende Namen (und Pfade) in Anweisungen "eingefügt" werden wird der Fachmann als Aktionen des Softwareentwicklers bei der Verwendung der beanspruchten Benutzeroberfläche verstehen.
- 5.2 Anspruch 1 der 1. und 2. Hilfsanträge fordert, dass ein "Anweisungsabschnitt, mit dem ein Schlüsselwort in der Anweisung ersetzt wird ... eine markierte Stelle aufweist, mit der die [einschlägige] Auswahlliste wieder aufgerufen werden kann". Der Anspruchswortlaut legt weder die Natur dieser "Stelle" fest noch in welcher Weise diese es ermöglichen solle, die Auswahlliste wieder aufzurufen. Damit würde der Fachmann nach Ansicht der Kammer dieses Merkmal funktional als die Forderung interpretieren, dass die einmal erfolgte Ersetzung eines Schlüsselwortes in einer Anweisung falls notwendig zur Korrektur wiederholt werden könne. Ob und in welcher Weise die grafische Benutzeroberfläche diese Funktion unterstützt, bleibt dabei offen.
- 5.3 Anspruch 1 der Hilfsanträge 3 und 4 fordert die Existenz von Objekten unterschiedlicher Kategorie, nämlich "Hauptobjekt", "Objekt" und "Subobjekt". Im Ladungszusatz (Punkt 7) bezweifelte die Kammer die Klarheit der einschlägigen Definition. Diese Frage kann jedoch für die vorliegende Entscheidung dahingestellt bleiben: Ob Containerobjekte (wie z. B. Listen und Arrays) und "einfache" Objekte (bspw. Integers) zur Verfügung gestellt werden und in welcher Weise Objekte andere Objekte oder Operationen "ergänzen", ist nach Ansicht der Kammer ein Aspekt der Definition der gewünschten Programmiersprache. Auch die Beschreibung offenbart ausdrücklich, dass es insbesondere von der "Mächtigkeit der Sprache" abhängt, welche Hauptobjekte, Objekte und

Subobjekte einem Softwareentwickler zur Verfügung stehen. Erneut ist festzustellen, dass der Anspruchswortlaut nicht festlegt, ob und in welcher Weise die grafische Benutzeroberfläche die Handhabung der unterschiedlichen Objektkategorien unterstützt.

*Technische Aufgabe und technischer Beitrag*

6. Die Anmeldung gibt als Ziele der vorliegenden Anmeldung an, die Programmierung von imperativen Programmen für Nicht-Spezialisten zu vereinfachen, sowie den so erzeugten Programmcode leichter lesbar und wartbar zu machen. Die Kammer ist der Ansicht, dass keines dieser Ziele als eine *technische* Aufgabe gelten kann, wie sie nach ständiger Rechtsprechung der Beschwerdekammern als Voraussetzung für eine erfinderische Tätigkeit im Sinne des Artikel 56 EPÜ 1973 unerlässlich ist. Die Kammer folgt in dieser Entscheidung ihren früheren Entscheidungen T 1539/09 und T 1171/06.
  
7. In T 1539/09 (1. Orientierungssatz) hat die Kammer festgestellt, dass die Tätigkeit des Programmierens als ein mentaler Vorgang anzusehen ist, soweit sie nicht im Rahmen einer konkreten Anwendung oder Umgebung in kausaler Weise der Erzielung einer technischen Wirkung dient.
  - 7.1 Eine solche Wirkung ist im vorliegenden Fall für die Kammer nicht ersichtlich. Insbesondere ist das beanspruchte Programmiersystem intentionsgemäß universell, nicht auf ein konkretes Anwendungsgebiet beschränkt und schließt ausdrücklich technische wie nicht-technische Anwendungsgebiete ein (vgl. S. 4, Zeilen 1-3). Ob und inwieweit die Erfindung daher die Programmentwicklung erleichtert, scheint der Kammer somit für die Bewertung der erfinderischen Tätigkeit nicht relevant zu sein.

- 7.2 Ob und inwieweit eine Programmiersprache und ein Programmiersystem das Programmieren erleichtert, hängt zum einen von vielen (nicht beschriebenen oder beanspruchten) Umständen ab und stellt zum anderen wenigstens zu einem erheblichen Teil ein subjektives Kriterium dar, das sich nur schwer, falls überhaupt, sinnvoll quantifizieren und verlässlich feststellen lässt. Die Anmeldung stellt keine belastbare Grundlage dar, auf der die behauptete Vereinfachung der Programmerstellung zu überprüfen wäre.
- 7.3 In der Anmeldung wird insbesondere dargestellt, dass die Erfindung eine neue Art des Programmierens - "[p]rozessuales Programmieren" - unterstütze, gestützt u. a. auf den "Grundgedanken", Programmanweisungen aus Operationen und Objekten" zu bilden und so "[d]ie Denkweise beim Programmieren ... in eine neue Richtung" zu lenken. Die Kammer verweist in dieser Hinsicht darauf, dass Programmiersprachen bekanntermaßen auf unterschiedlichen sogenannten "Paradigmen" beruhen, die jeweils eine unterschiedliche Sicht auf die zu entwickelnden Programme ermöglichen oder sogar erzwingen: Beispielsweise stehen im Zentrum "objekt-orientierter" Programme sogenannte "Objekte", die jeweils die "Methoden" zu ihrer Bearbeitung gleich selbst bereitstellen, und Lösungen in objekt-orientierten Sprachen müssen als Kooperation solcher, miteinander kommunizierender Objekte konzipiert werden; im Zentrum "funktionaler" Programme stehen hingegen mathematische Ausdrücke, und Lösungen in solchen Sprachen müssen als die Auswertung bzw. Vereinfachung solcher Ausdrücke konzipiert werden.
- 7.4 Die Bereitstellung der entsprechenden Programmiersprache und ihrer Ausdrucksmittel wie Objekt/Methode/Nachricht, Funktion/Auswertung oder eben Operation/Objekt wie im vorliegenden Fall trägt nach Ansicht der

Kammer - und gemäß T 1539/09 (2. Orientierungssatz) - nicht zur Lösung eines technischen Problems bei.

8. In T 1171/06 (Orientierungssatz) stellte die Kammer weiter fest, dass einem in der Softwareentwicklung verwendeten Modell kein technischer Effekt dadurch zukomme, dass es der Dokumentation oder Kommunikation dient, selbst wenn sein Gegenstand ein technisches System sei.
  - 8.1 In diesem Sinne hält die Kammer auch die behauptete Erleichterung der Lesbarkeit und Wartbarkeit der erstellten Programme für die Bewertung der erfinderischen Tätigkeit nicht für relevant.
  - 8.2 Auch hier sei aber zudem angemerkt, dass die Frage, ob ein Programm leicht lesbar und wartbar ist, eine großteils subjektive ist, und es schwierig bis unmöglich zu sein scheint, einen solchen Schwierigkeitsgrad sinnvoll und verlässlich zu beziffern.
9. Die Kammer schließt nicht aus, dass konkrete Details einer graphischen Benutzeroberfläche die Bedienung eines Computers als ein technisches Gerät und/oder bei der Anwendung auf eine technische Aufgabe erleichtern, und auf diese Weise einen Beitrag zur Lösung eines technischer Aufgabe dienen können.
  - 9.1 Allerdings legt wenigstens Anspruch 1 aller vorliegenden Anträge - wie oben ausgeführt - allenfalls fest, dass die Programmierung mit der vorgeschlagenen Programmiersprache durch eine grafische Benutzeroberfläche unterstützt wird, nicht aber durch welche konkreten Mittel.

- 9.2 Im Ladungszusatz informierte die Kammer die Beschwerdeführerin darüber, dass es ihr nicht ersichtlich sei, welche konkreten Merkmale der graphischen Benutzeroberfläche gemäß einem der vorliegenden Anträge einen technischen Beitrag leisten würden.
- 9.3 Die Beschwerdeführerin hat auf diese Feststellung der Kammer nicht reagiert, also weder Argumente vorgebracht noch Änderungen eingereicht, aus denen der technische Beitrag eines beanspruchten Merkmals der grafischen Benutzeroberfläche hervorgehen würde.
- 9.4 Daher hat die Kammer keinen Anlass, von ihrer vorläufigen Meinung abzuweichen.
10. Zusammenfassend also interpretiert die Kammer den Gegenstand von Anspruch 1 aller vorliegenden Anträge als ein computergestütztes Verfahren, demgemäß die Programmierung einer gewünschten imperativen Programmiersprache durch eine im Wesentlichen undefinierte grafische Benutzerschnittstelle unterstützt wird.
- 10.1 In Übereinstimmung mit T 1539/09 ist die Kammer der Ansicht, dass die Definition der gewünschten Programmiersprache keine technische Aufgabe löst und somit keinen Beitrag zur erfinderischen Tätigkeit leisten kann. Im Rahmen des Aufgabe-Lösungs-Ansatzes dürfen die entsprechenden Merkmale somit, gemäß T T641/00 (ABl. EPA, 2003, 352; Orientierungssatz II), bei der Formulierung der Aufgabe als Teil der Rahmenbedingungen für die zu lösende technische Aufgabe aufgegriffen werden, insbesondere als eine zwingend zu erfüllende Vorgabe.
- 10.2 Die von der Erfindung gemäß Anspruch 1 aller Anträge gelöste Aufgabe kann somit darin gesehen werden, die

Entwicklung von Programmen in einer Programmiersprache mit den geforderten Merkmalen zu unterstützen.

- 10.3 Programmierumgebungen mit grafischen Benutzeroberflächen sind zu diesem Zwecke allgemein bekannt und daher ohne Weiteres naheliegend.
- 10.4 Daher kommt die Kammer zu dem Schluss, dass der Gegenstand von Anspruch 1 aller Anträge schon gegenüber dem allgemeinen Fachwissen keine erfinderische Tätigkeit im Sinne von Artikel 56 EPÜ 1973 aufweist.
11. Die im Ladungszusatz darüber hinaus aufgeworfene Frage, ob und warum im Einzelnen es dem beanspruchte Gegenstand außerdem gegenüber D1 an einer erfinderischen Tätigkeit mangelt, kann somit dahingestellt bleiben.

### **Entscheidungsformel**

#### **Aus diesen Gründen wird entschieden:**

Die Beschwerde wird zurückgewiesen.

Die Geschäftsstellenbeamtin:

Der Vorsitzende:



B. Atienza Vivancos

D. Rees

Entscheidung elektronisch als authentisch bestätigt