

(12) NACH DEM VERTRAG ÜBER DIE INTERNATIONALE ZUSAMMENARBEIT AUF DEM GEBIET DES PATENTWESENS (PCT) VERÖFFENTLICHTE INTERNATIONALE ANMELDUNG

(19) Weltorganisation für geistiges Eigentum
Internationales Büro



(43) Internationales Veröffentlichungsdatum
30. August 2007 (30.08.2007)

PCT

(10) Internationale Veröffentlichungsnummer
WO 2007/096244 A1

(51) Internationale Patentklassifikation:
G06Q 10/00 (2006.01)

(21) Internationales Aktenzeichen: PCT/EP2007/051081

(22) Internationales Anmeldedatum:
5. Februar 2007 (05.02.2007)

(25) Einreichungssprache: Deutsch

(26) Veröffentlichungssprache: Deutsch

(30) Angaben zur Priorität:
10 2006 007 791.1
20. Februar 2006 (20.02.2006) DE

(71) Anmelder (für alle Bestimmungsstaaten mit Ausnahme von US): **SIEMENS AKTIENGESELLSCHAFT** [DE/DE];
Wittelsbacherplatz 2, 80333 München (DE).

(72) Erfinder; und

(75) Erfinder/Anmelder (nur für US): **HENNIG, Andreas**

[DE/DE]; Winningerstr. 7, 81737 München (DE).
OLKHOVITCH, Lev [RU/DE]; Eichendorffstrasse 17,
85521 Ottobrunn (DE). **WASGINT, Rainer** [DE/DE];
Carl-Baierl-Bogen 12, 81739 München (DE).

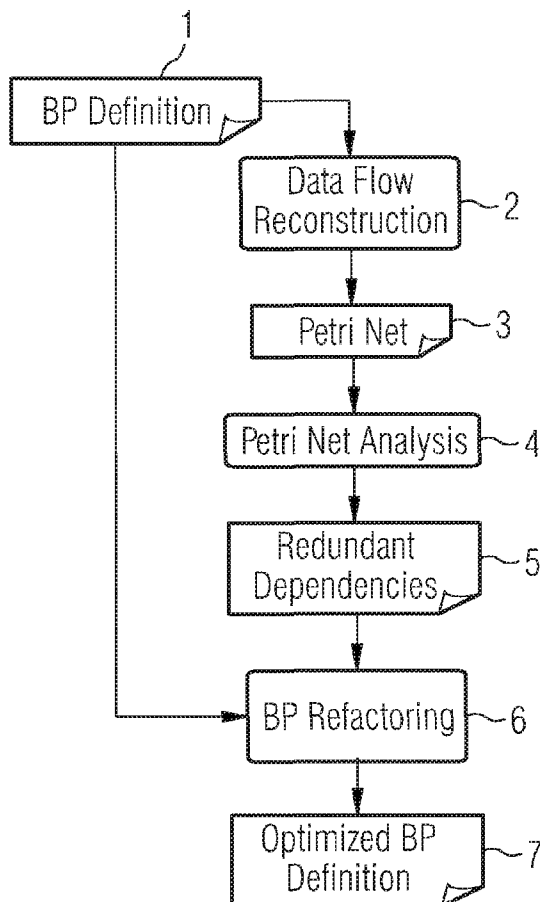
(74) Gemeinsamer Vertreter: **SIEMENS AKTIENGESELLSCHAFT**; Postfach 22 16 34, 80506 München (DE).

(81) Bestimmungsstaaten (soweit nicht anders angegeben, für jede verfügbare nationale Schutzrechtsart): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LV, LY, MA, MD, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, SV, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

[Fortsetzung auf der nächsten Seite]

(54) Title: METHOD FOR PROCESS OPTIMISATION

(54) Bezeichnung: VERFAHREN ZUR PROZESSOPTIMIERUNG



(57) Abstract: The invention is essentially characterised in that firstly, the process as described by a process definition is analysed such that those control sequence dependencies are discovered which are not supported by the data stream, the control sequences contained in the process are transformed in a first step into a corresponding Petri network and, in a second step, said Petri network is analysed. Secondly, the process is then reconstructed without the discovered redundant or unnecessary control sequence dependencies and an optimum process definition generated. Excess dependencies are thus automatically discovered and removed by means of the invention, in order to achieve, for example, an increased parallelisation or compatibility of partial processes.

(57) Zusammenfassung: Die Erfindung besteht im Wesentlichen darin, dass als erstes der durch eine Prozessdefinition beschriebene Prozess derart analysiert wird, dass Steuerablauf-Abhängigkeiten aufgedeckt werden die nicht durch den Datenfluss unterstützt sind, wobei der im Prozess enthaltene Steuerablauf in einem ersten Schritt in ein entsprechendes Petri-Netz umgeformt wird und dann in einem zweiten Schritt dieses Petrinetz analysiert wird. Als zweites wird dann der Prozess ohne die aufgedeckten redundanten bzw. unnötigen Steuerablauf-Abhängigkeiten rekonstruiert und eine optimierte Prozessdefinition erstellt. Durch die Erfindung werden so übermäßig vorhandene Abhängigkeiten automatisch aufgefunden und beseitigt, um z.B. eine erhöhte Parallelisierung bzw. Nebenläufigkeit von Teilprozessen zu erreichen.

WO 2007/096244 A1



(84) **Bestimmungsstaaten** (soweit nicht anders angegeben, für jede verfügbare regionale Schutzrechtsart): ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), eurasisches (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), europäisches (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Veröffentlicht:

— mit internationalem Recherchenbericht

Zur Erklärung der Zweibuchstaben-Codes und der anderen Abkürzungen wird auf die Erklärungen ("Guidance Notes on Codes and Abbreviations") am Anfang jeder regulären Ausgabe der PCT-Gazette verwiesen.

Beschreibung

Verfahren zur Prozessoptimierung

- 5 Die Erfindung betrifft ein Verfahren zur Optimierung eines Prozesses, bspw. eines Netzdienstes (web services), durch Beseitigung übermäßig vorhandener Abhängigkeiten zwischen einzelnen Aktivitäten effektivere Prozesse gebildet werden.
- 10 Ineffektive Prozesse führen zu exzessiven Richtungsänderungen, Wartezeiten und zu unausgelasteten bzw. überlasteten Ressourcen.

Eine Prozessoptimierung erfolgt momentan manuell und
15 basierend auf Intuition und Erfahrung. Es sind lediglich einige quantitative Analysetechniken, wie bspw. Jonkers, H.; and H.M. Franken. 1996. "*Quantitative modelling and analysis of business processes*", in A. Bruzzone and E. Kerckhoffs, eds., *Simulation in Industry: Proceedings 8 th European Simulation Symposium, vol. I, Genoa, Italy, Oct., 175-179*; und Datenfluss-Verifikationstechniken, wie z. B.
20 <http://crpit.com/confpapers/CRPITV27Sadiq.pdf>
bekannt.

25 Aus der Veröffentlichung von W.M.P van der Aalst mit dem Titel „The Application of Petri Nets to Workflow Management“, <http://is.tm.tue.nl/staff/wvdaalst/publications/p53.pdf> ist die Abbildung von Geschäftsprozessen auf Petri-Netze bekannt.

30

Die der Erfindung zu Grunde liegende Aufgabe besteht nun darin ein Verfahren zur Prozessoptimierung anzugeben, bei dem übermäßig vorhandene Abhängigkeiten automatisch aufgefunden und beseitigt werden.

35

Dies wird erfindungsgemäß durch die Merkmale des Patentanspruchs 1 gelöst. Die weiteren Patentansprüche

betreffen bevorzugte Ausgestaltungen des erfindungsgemäßen Verfahrens.

Die Erfindung besteht im Wesentlichen darin, dass als erstes
5 der durch eine Prozessdefinition beschriebene Prozess derart
analysiert wird, dass Steuerablauf-Abhängigkeiten aufgedeckt
werden die nicht durch den Datenfluss unterstützt sind, wobei
der im Prozess enthaltene Steuerablauf in einem ersten
Schritt in ein entsprechendes Petri-Netz umgeformt wird und
10 dann in einem zweiten Schritt dieses Petrinetz analysiert
wird. Als zweites wird dann der Prozess ohne die
aufgedeckten redundanten bzw. unnötigen Steuerablauf-
Abhängigkeiten rekonstruiert und eine optimierte
Prozessdefinition erstellt. Durch die Erfindung werden so
15 übermäßig vorhandene Abhängigkeiten automatisch aufgefunden
und beseitigt, um z. B. eine erhöhte Parallelisierung bzw.
Nebenläufigkeit von Teilprozessen zu erreichen.

Nachfolgend wird die Erfindung anhand eines in der Zeichnung
20 dargestellten Ausführungsbeispiels näher erläutert. Dabei
zeigt

- Figur 1 eine Prinzipdarstellung zur Erläuterung des
erfindungsgemäßen Verfahrens,
25
- Figur 2 einen Aktivitäts-Graph eines beispielhaften
Original-Prozesses zusammen mit einer Umsetzung in
eine entsprechendes Petri-Netz-Darstellung,
- 30 Figuren 4A, 4B, 5A, 5B, 6A, 6B Petri-Netz-Darstellungen
zur Erläuterung von Besonderheiten bei der
Umsetzung eines Aktivitäts-Graphen in ein
entsprechendes Petri-Netz und
- 35 Figur 7 einen Aktivitäts-Graph des durch das
erfindungsgemäße Verfahren optimierten
beispielhaften Original-Prozesses.

In **Figur 1** ist eine Prinzipdarstellung zur Erläuterung des erfindungsgemäßen Verfahrens dargestellt, bei der ausgehend von einer Prozessdefinition 1, bspw. in Form eines Aktivitäts-Graphen (activity graph) oder sog. BPEL, eine Rekonstruktion 2 eines zugehörigen Datenflusses erfolgt und ein entsprechendes Petrinetz 3 gebildet wird, das einer nachfolgenden Analyse 4 des Petrinetzes unterzogen wird, um redundante Abhängigkeiten 5 zwischen einzelnen Aktivitäten zu ermitteln. Stehen diese redundanten Abhängigkeiten fest, wird mit dieser Information aus der ursprünglichen Prozessdefinition 1 ein Umbau (process refactoring) 6 zu einer Prozessdefinition 6 des optimierten Prozesses durchgeführt.

In **Figur 2** ist ein Aktivitäts-Graph eines Original-Prozesses zusammen mit einer Umsetzung in ein entsprechendes Petri-Netz am Beispiel eines Bahnticket-Verkaufsautomaten gezeigt. Im Aktivitäts-Graph sind vom Start bis zum Ende die einzelnen Aktivitäten C1 ...C9 über Pfeile, z. B. C1C2, im Steuerablaufplan verbunden und daneben sind Daten D1...D6 angeordnet, wobei durch gestrichelte Daten-Pfeile ersichtlich ist durch welche Aktionen diese Daten gebildet und von welchen diese Daten gebraucht bzw. entgegengenommen werden. Der rechte Teil von **Figur 2** zeigt einen Ausschnitt eines entsprechenden Petri-Netzes für die Aktivitäten C4..C7 und die Daten D3..D5, wobei die Aktivitäten C4..C7 Transitionen T1..T4 im Petri-Netz entsprechen. Wie bei Petri-Netzen üblich ist auch hier zwischen Transitionen mindestens je ein Platz 1..3 vorhanden.

Bei der Bildung des Petri-Netzes werden, wie in **Figur 3** gezeigt, Aktivitäten die auf keine Daten zugreifen, Aktivitäten zum Schreiben von bisher nicht initialisierten Daten, Aktivitäten die Daten lesen, siehe **Figuren 4A und 4B**, und Aktivitäten die Daten modifizieren, siehe **Figuren 5A und 5B**, unterschieden. Für den Fall dass eine Folgeaktivität sowohl Daten im Originalzustand aber auch im modifizierten Zustand akzeptiert, wird, wie dies **Figuren 6A und 6B** gezeigt

ist, im Petri-Netz durch je einen Platz vor und nach der Aufspaltung berücksichtigt. Es können auch Hilfstransitionen und Hilfsplätze eingefügt werden, um bspw. eine Änderung der Daten während einer Aktion zu ermöglichen. Diese Hilfs-
5 Transitionen und Hilfsplätze sind in den **Figuren 4B, 5B und 6B** schraffiert dargestellt. Das Problem, dass, sofern zwei Aktionen auf die selben Ressourcen zurückgreifen, normalerweise keine strenge Ausführungsreihenfolge gewährleistet werden kann, wird entweder dadurch gelöst, dass
10 parallele Aktivitäten die ihre eigenen Daten speichern oder zwei getrennte Ressourcen verwendet werden. Die Analyse des Petri-Netzes stellt zunächst die Abhängigkeiten zwischen den Einzelaktivitäten fest und vergleicht dann den Steuerablauf (control flow) des Prozesses mit diesen Abhängigkeiten.

15

Hierzu wird, wie auch aus dem Pseudocode Listing1 im Anhang 1 ersichtlich, für jede Transition t eine Menge SET(t) von anderen Transitionen ermittelt, von denen die Transition t abhängt.

20

Im Anhang 3 wird für das konkrete Beispiel des Bahnticketverkaufsautomaten der Programmablauf (Trace) des Pseudocodes Listing1 mitverfolgt bzw. dokumentiert, wobei Variablennamen fettgedruckt und Petri-Netz-Elemente kursiv gedruckt sind.

25

Danach wird, wie auch aus dem weiteren Pseudocode Listing2 im Anhang 2 ersichtlich, der Steuerablauf (control flow) den Abhängigkeiten zwischen den in der Menge SET enthaltenen
30 Einzelaktivitäten gegenübergestellt und verglichen. Verbindungspfeile im Steuerablaufplan, die durch diese Abhängigkeiten gestützt sind bzw. gefordert werden, werden in einer zweiten Menge SET2 gespeichert. Hierzu wird eine vorteilhafter Weise eine Verbindungsüberprüfungs-Funktion
35 CHECK-CONNECTION definiert und verwendet, die alle Pfeile zwischen einer Quelle SOURCE und einem Ziel TARGET dahingehend überprüft, ob ein Pfeil durch mindestens eine Daten-Abhängigkeit zwischen SOURCE und TARGET gestützt ist.

Im Anhang 4 wird für das konkrete Beispiel des Bahnticketverkaufsautomaten der Programmablauf des Pseudocodes Listing2 mitverfolgt bzw. dokumentiert, wobei
5 wiederum Variablennamen fettgedruckt und Petri-Netz-Elemente kursiv gedruckt sind.

In **Figur 7** ist ein Aktivitäts-Graph des durch das erfindungsgemäße Verfahren optimierten beispielhaften
10 Original-Prozesses gezeigt, wobei hier deutlich wird, dass die Aktion C5 „Read EC-Card“ nämlich das Lesen der EC-Karte nicht erst nach der Aktion C4 sondern praktisch gleichzeitig mit
mit
der Aktion C2 „Input Search Criteria“ erfolgen kann, da die
15 Aktion C5 zum Lesen der EC-Karte die aus den Aktionen C2..C4 erst gebildeten Suchkriterien D1, den Zugfahrplan D2 und die Zugdaten D3 gar nicht benötigt. Mit der Aktion C7 „Payment Process“ erfolgt eine Zusammenführung bzw. Synchronisation der Aktionen, wobei hier zu bemerken ist, dass die PIN-
20 Eingabe aus Sicherheitsgründen unmittelbar vor dem eigentlichen Zahlungsvorgang erfolgt.

Im ursprünglich Prozess werden die Aktionen der Zugauswahl und das Lesen der EC-Karte sequentiell durchgeführt und es
25 addieren sich somit die Verarbeitungszeiten zu $T=t_1+t_2$, wohingegen beim optimierten Prozess zu $T_{opt}=\max(t_1,t_2)$; der Zeitgewinn beträgt also $\Delta T= \text{Abs}(t_1-t_2)$.

Anhang 1

Listing 1

```
5  1.Take all the transitions that have no incoming data arcs and put
   them into transition queue.
   2.While transition queue not empty:
       1.Fetch a transition (START) from the transition queue
       2.For every data arc (ARC) originating from START:
10      1.If ARC's target has data arcs targeting START:
           Continue to new iteration.
           End If
       2.For every data arc (ARC2) originating at the ARC's target:
           1.Add START to the SET( ARC2's target transition ).
15      2.If ARC2's target not yet processed, add it to the
           transition queue.
```


Anhang 2

Listing 2

```

1. Take all the transitions directly connected to the initial place
5 and put them into transition queue.
2. While transition queue not empty:
    1. Fetch a transition TRANS from the transition queue.
    2. For every control place (PLACE) directly reachable from TRANS:
        1. If PLACE has not been processed before:
10         For each control arc ARC originating at PLACE:
            If CHECK-CONNECTION (TRANS, ARC's target TRANS2) is TRUE:
                Add all the control arcs connecting TRANS
                to TRANS2 to SET2.
            End If
15         End If
Boolean Function CHECK-CONNECTION (Transitions SOURCE, TARGET):
1. If TARGET is a split/join:
    1. For every control place PLACE2 directly reachable from SOURCE:
        If PLACE has not been processed before:
20         For each control arc ARC originating at PLACE:
            If CHECK-CONNECTION (TARGET, ARC's target TRANS) is TRUE:
                Add all the control arcs connecting TARGET
                to TRANS to SET2.
            End If
25         End If
    2. If CHECK-CONNECTION ever returned TRUE in the loop above:
        Return TRUE.
    Else:
        Return FALSE.
30 2. Else:
    1. Add TARGET to the transition queue.
    2. If SET (TARGET) contains SOURCE:
        Return TRUE.
    Else:
35         Return FALSE.

```

Anhang 3

Pseudocode 1 - Trace für das konkrete Anwendungsbeispiel

Variablennamen sind fettgedruckt und Petrinetzelement-Namen sind kursiv gedruckt.

```

1. Transition queue = {"Select Train", "Read EC Card"}
  ⇒ 2.1: START = "Select Train"
  ⇒ 2.2: Outgoing data arcs: "Select Train → Train Data"

  [1] ARC = "Select Train → Train Data"
    ⇒ 2.2.1: "Train Data" has no data arcs targeting "Select Train"
    ⇒ 2.2.2: Data arcs originating at "Train Data" are:
      "Train Data → Confirm/Input PIN", "Train Data → Process Payment"

      [1] ARC2 = "Train Data → Confirm/Input PIN":
        ⇒ 2.2.2.1: "Select Train" is added to SET("Confirm/Input PIN"),
          SET("Confirm/Input PIN") = {"Select Train"}
        ⇒ 2.2.2.2: "Confirm/Input PIN" is added to the transition queue,
          transition queue = {"Read EC Card", "Confirm/Input PIN"}

      [2] ARC2 = "Train Data → Process Payment" :
        ⇒ 2.2.2.1 "Select Train" is added to SET("Process Payment"),
          SET("Process Payment") = {"Select Train"}
        ⇒ 2.2.2.2: "Select Train" is added to transition queue,
          transition queue = {"Read EC Card", "Confirm/Input PIN", "Process
          Payment"}

2. Transition queue = {"Read EC Card" , "Confirm/Input PIN", "Process Payment"}
  ⇒ 2.1: START = "Read EC Card"
  ⇒ 2.2: Outgoing data arcs: "Read EC Card → EC Card Data"

  [1] ARC = "Read EC Card → EC Card Data"
    ⇒ 2.2.1: "EC Card Data" has no data arcs targeting "Read EC Card"
    ⇒ 2.2.2: Data arcs originating at "EC Card Data" are:
      "Card Data → Process Payment"

      [1] ARC2 = "Card Data → Process Payment"
        ⇒ 2.2.2.1: "Read EC Card" is added to SET("Process Payment"),
          SET("Process Payment") = {"Select Train", "Read EC Card"}

```

⇒ 2.2.2.2: "Process Payment" was already processed

3. **Transition queue** = {"Confirm/Input PIN" , "Process Payment"}

⇒ 2.1: **START** = "Confirm/Input PIN"

⇒ 2.2: Outgoing data arcs: "Confirm/Input PIN → Train Data" and
"Confirm/input PIN → PIN"

[1] **ARC** = "Confirm/Input PIN → Train Data"

⇒ "Train Data" has arcs targeting "Confirm/Input PIN", continue to next iteration

[2] **ARC** = "Confirm/Input PIN → PIN"

⇒ 2.2.1: "PIN" has no arcs targeting "Confirm/Input PIN"

⇒ Data arcs originating at "PIN" are: "PIN → Process Payment"

[1] **ARC2** = "PIN → Process Payment"

⇒ 2.2.2.1: "Confirm/Input PIN" is added to **SET**("Process Payment"),
SET("Process Payment") = {"Select Train", "Read EC Card",
"Confirm/input PIN"}

⇒ 2.2.2.2: "Process Payment" was already processed

4. **Transition queue** = {"Process Payment"}

⇒ 2.1: **START** = "Process Payment"

⇒ 2.2: Outgoing data arcs: "Process Payment → Train Data ", "Process Payment →
Card Data ", "Process Payment → PIN"

[1] **ARC** = "Process Payment → Train Data"

⇒ "Train Data" has arcs targeting "Process Payment", continue to next iteration

[2] **ARC** = "Process Payment → Card Data"

⇒ "Card Data" has arcs targeting "Process Payment", continue to next iteration

[3] **ARC** = "Process Payment → PIN"

⇒ "PIN" has arcs targeting "Process Payment", continue to next iteration

5. **Transition queue** = {}

Anhang 4

Pseudocode 2 - Trace für das konkrete Anwendungsbeispiel

Variablennamen sind fettgedruckt und Petrinetzelement-Namen sind kursiv gedruckt.

```

1. Transition queue = {"Select Train" }
  ⇒ 2.1: START = "Select Train"
  ⇒ 2.2: Outgoing control arcs: "Select Train" → 1

  [1] PLACE = "1", it has not been processed before
    ⇒ 2.2.1: Control arcs originating at "1" are: "1 → Read EC Card"

      [1] ARC = "1 → Read EC Card":
        ⇒ 2.2.1.1: CHECK-CONNECTION ( SOURCE = "Select Train", TARGET = "Read EC Card" ) == FALSE:

          ∇ 1: "Read EC Card" is not a split/join
            ⇒ 2.1: "Read EC Card" is added to transition queue,
              transition queue = {"Read EC Card"}
            ⇒ 2.2: SET("Read EC Card") does not contain "Select Train",
              return FALSE

2. Transition queue = {"Read EC Card"}
  ⇒ 2.1: START = "Read EC Card"
  ⇒ 2.2: Outgoing control arcs: "Read EC Card" → 2

  [1] PLACE = "2", it has not been processed before
    ⇒ 2.2.1: Control arcs originating at "2" are: "2 → Confirm/Input PIN"

      [1] ARC = "2 → Confirm/Input PIN"
        ⇒ 2.2.1.1: CHECK-CONNECTION ( SOURCE = "Read EC Card",
          TARGET = Confirm/Input PIN ) == FALSE:

          ∇ 1: "Confirm/Input PIN" is not a split/join
            ⇒ 2.1: "Confirm/Input PIN" is added to transition queue,
              transition queue = {"Confirm/Input PIN"}
            ⇒ 2.2: SET("Confirm/Input PIN") does not contain "Read EC Card",
              return FALSE

```

3. **Transition queue** = {"Confirm/Input PIN" }
- ⇒ 2.1: **START** = "Confirm/Input PIN"
- ⇒ 2.2: Outgoing control arcs: "Confirm/Input PIN → 3"
- [1] **PLACE** = "3"
- ⇒ 2.2.1: Control arcs originating at "PIN" are: "3 → Process Payment"
- [1] **ARC** = "3 → Process Payment"
- ⇒ 2.2.1.1: CHECK-CONNECTION (**SOURCE** = "Confirm/Input PIN",
TARGET = "Process Payment") == TRUE,
arcs "Confirm/Input PIN → 3" and "3 → Process Payment" are added
to **SET2**, **SET2** = {"Confirm/Input PIN → 3", "3 → Process Payment"}
- ∇ 1: "Process Payment" is not a split/join
- ⇒ 2.1: "Process Payment" is added to **transition queue**,
transition queue = {"Confirm/Input PIN"}
- ⇒ 2.2: **SET**("Process Payment") does contain "Confirm/Input
PIN", return TRUE
4. **Transition queue** = {"Process Payment"}
- ⇒ 2.1: **START** = "Process Payment"
- ⇒ 2.2: Outgoing data arcs: none
5. **Transition queue** = {}

Patentansprüche

1. Verfahren zur Prozessoptimierung,

- bei dem als erstes ein durch eine Prozessdefinition (1)
5 beschriebener zu optimierender Prozess derart analysiert
wird, dass im Steuerablauf des Prozesses Steuerablauf-
Abhängigkeiten aufgedeckt werden die nicht durch einen
entsprechenden aus der Prozessdefinition rekonstruierten
10 Datenfluss (2) unterstützt sind, wobei der Steuerablauf des
Prozesses in ein entsprechendes Petri-Netz (3) umgeformt wird
und dieses Petrinetz mit Hilfe einer Petri-Netz-Analyse (4)
dann analysiert wird und
- bei dem der Prozess ohne die aufgedeckten überflüssigen
Steuerablauf-Abhängigkeiten (5) mit Hilfe der
15 Prozessdefinition (1) rekonstruiert und eine optimierte
Prozessdefinition (7) erstellt wird.

2. Verfahren nach Anspruch 1,

- bei dem bei der Petri-Netz-Analyse (4) für jede Transition t
20 eine Menge von anderen Transitionen ermittelt wird, von denen
die jeweilige Transition t abhängt.

3. Verfahren nach Anspruch 2,

- bei dem der Steuerablauf des Prozesses den Abhängigkeiten
25 zwischen den in dieser Menge enthaltenen Einzelaktivitäten
gegenübergestellt und verglichen wird und Verbindungspfeile
(C1C2) eines den Steuerablauf beschreibenden
Steuerablaufplans, die durch diese Abhängigkeiten gestützt
sind bzw. gefordert werden in einer zweiten Menge gespeichert
30 werden.

FIG 1

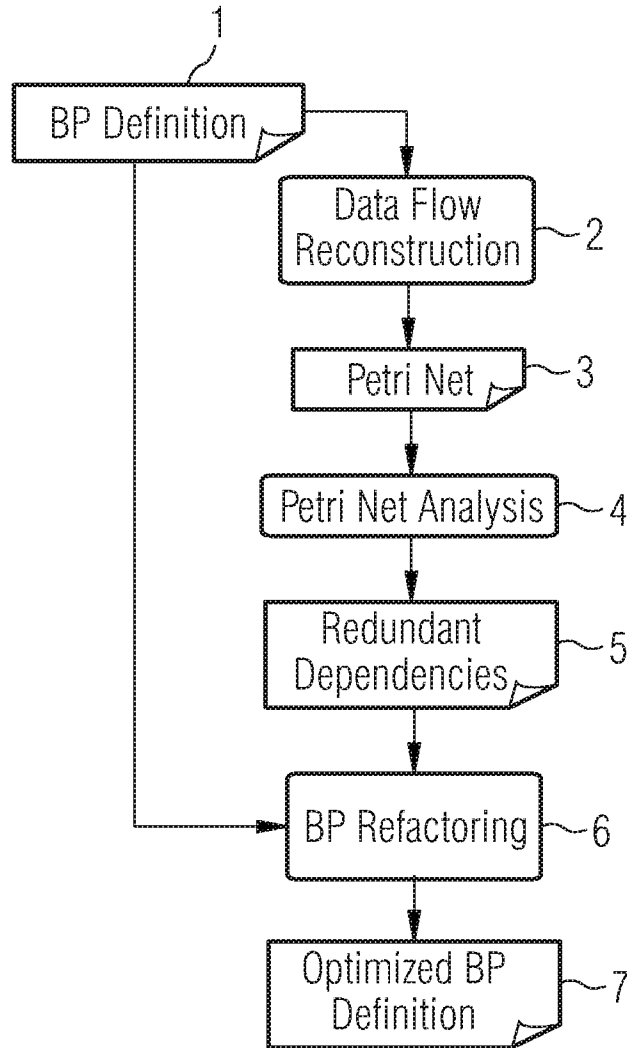


FIG 2

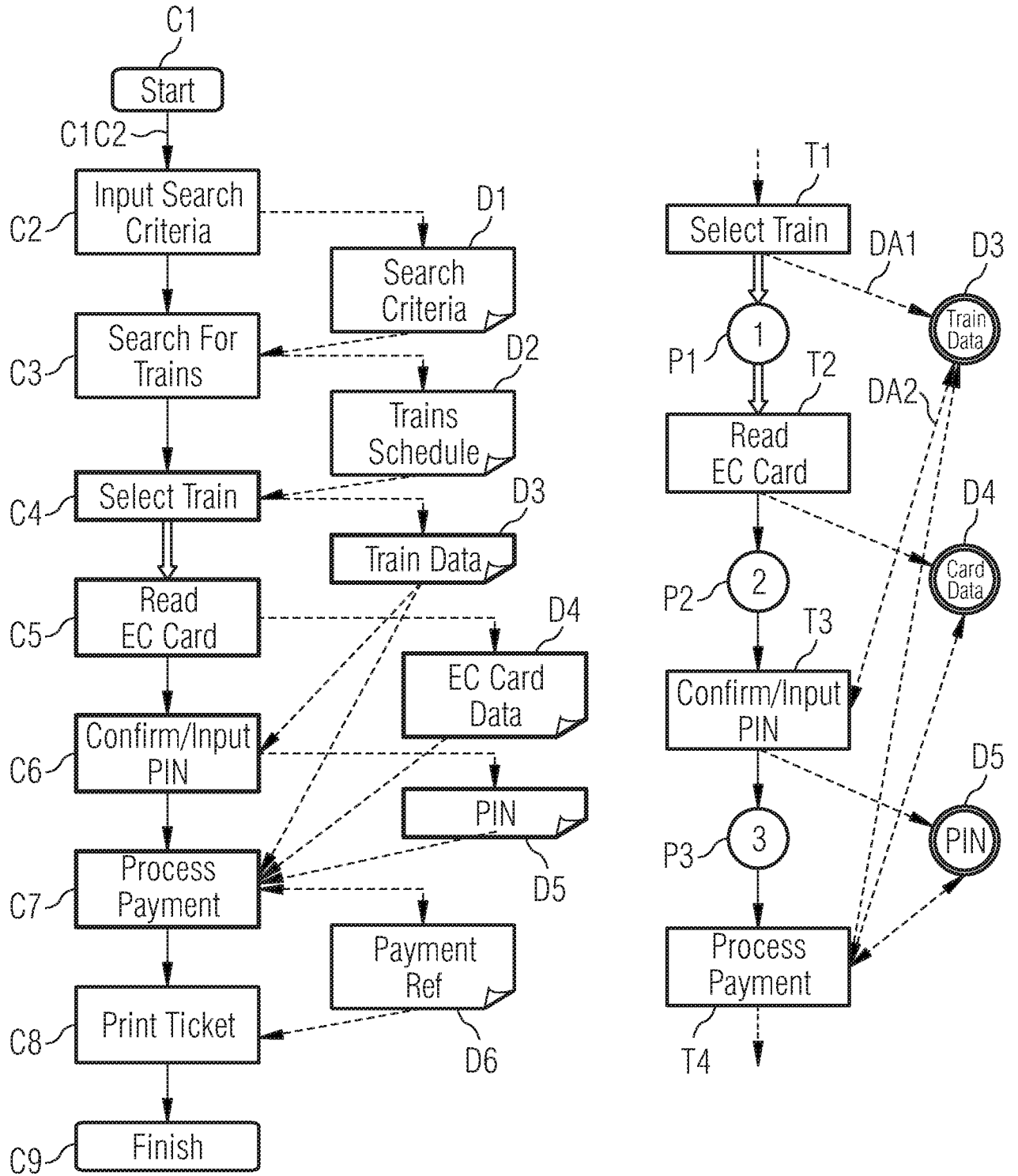


FIG 3

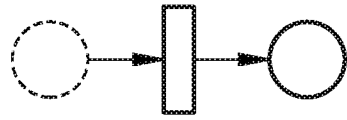


FIG 4A

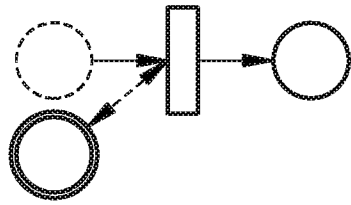


FIG 4B

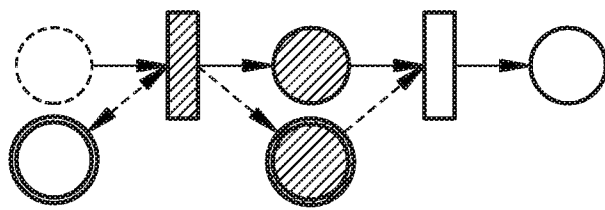


FIG 5A

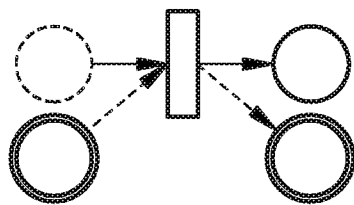


FIG 5B

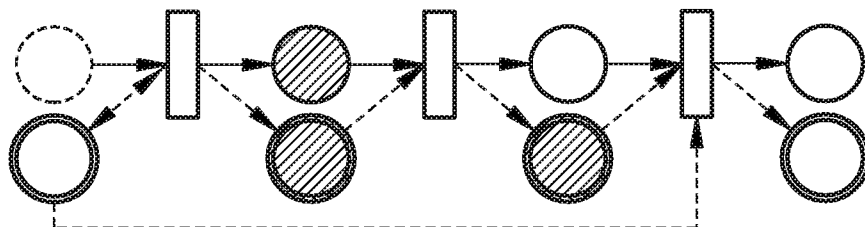


FIG 6A

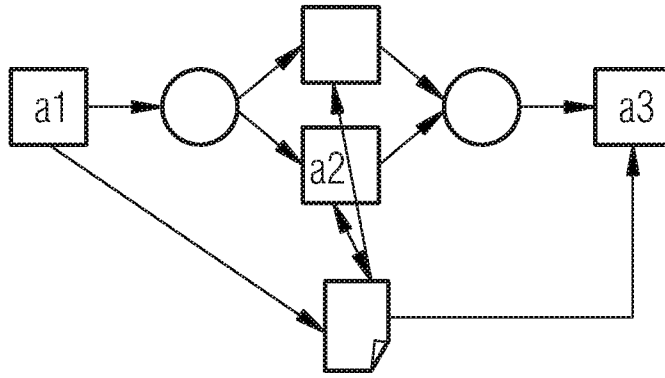


FIG 6B

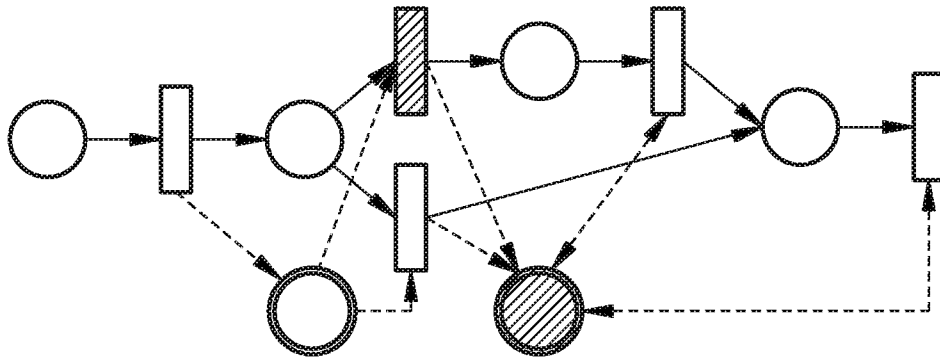
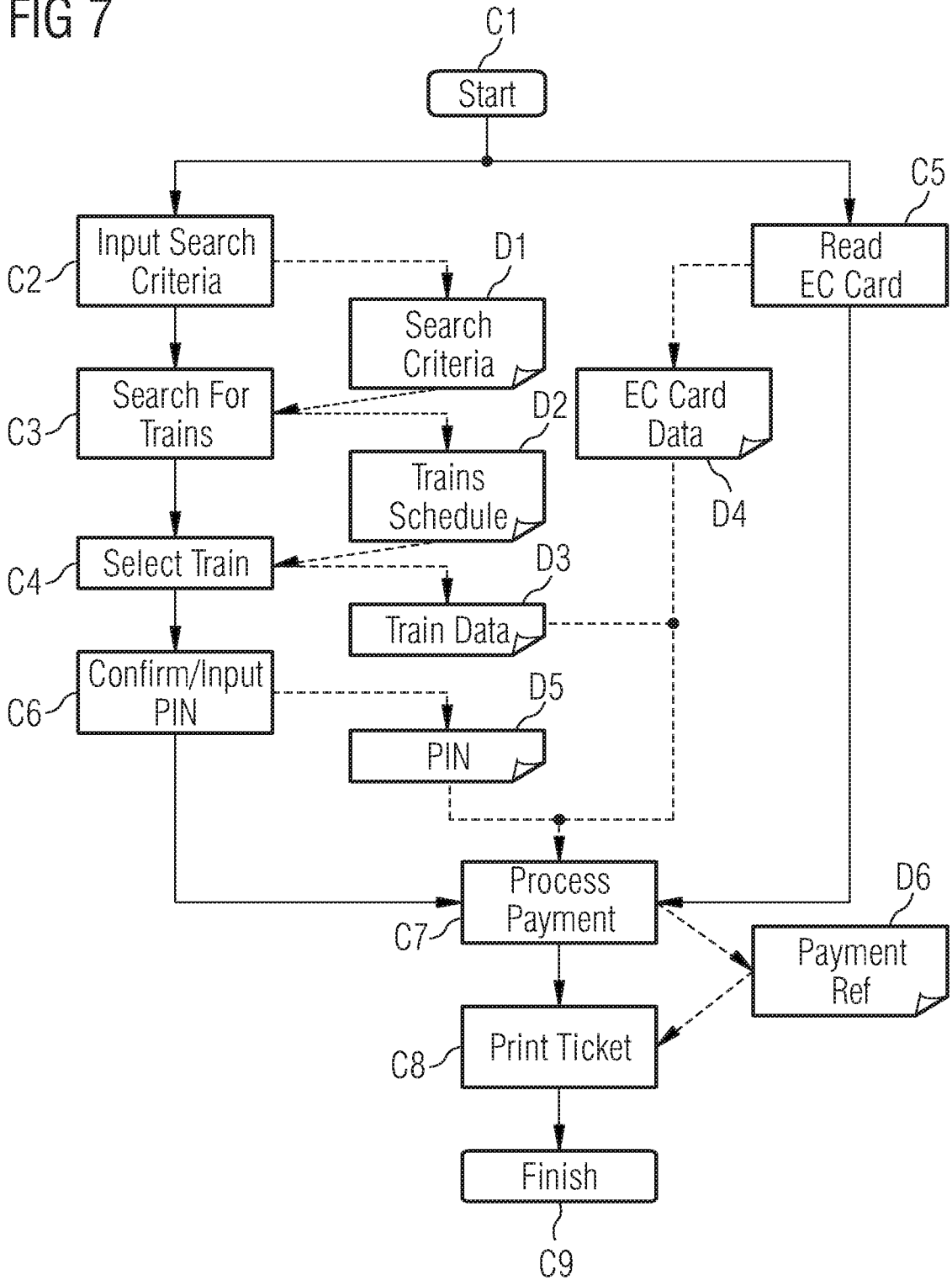


FIG 7



INTERNATIONAL SEARCH REPORT

International application No
PCT/EP2007/051081

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06Q10/00

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06Q

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

EPO-Internal

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	<p>OLKHOVICH L: "Semi-Automatic Business Process Performance Optimization Based On Redundant Control Flow Detection" TELECOMMUNICATIONS, 2005. AICT-ICIW '06. INTERNATIONAL CONFERENCE ON INTERNET AND WEB APPLICATIONS AND SERVICES/ADVANCED INTERNATIONAL CONFERENCE ON GUADELOPE, FRENCH CARIBBEAN 19-25 FEB. 2006, PISCATAWAY, NJ, USA, IEEE, 19 February 2006 (2006-02-19), pages 146-146, XP010898469 ISBN: 0-7695-2522-9 the whole document</p> <p style="text-align: center;">----- -/--</p>	1-3

Further documents are listed in the continuation of Box.C.

See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search

24 May 2007

Date of mailing of the international search report

12/06/2007

Name and mailing address of the ISA/

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,
Fax: (+31-70) 340-3016

Authorized officer

Streit, Stefan

INTERNATIONAL SEARCH REPORT

International application No
PCT/EP2007/051081

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	SEBASTIAN HINZ ET AL: "Transforming BPEL to Petri Nets. In: BPM 2005." LECTURE NOTES IN COMPUTER SCIENCE, vol. 3649, 2005, pages 220-235, XP019017192 the whole document	1-3
A	AALST VAN DER W M P: "THE APPLICATION OF PETRI NETS TO WORKFLOW MANAGEMENT" JOURNAL OF CIRCUITS, SYSTEMS AND COMPUTERS, WORLD SCIENTIFIC, LONDON, GB, vol. 8, no. 1, February 1998 (1998-02), pages 21-66, XP008038460 ISSN: 0218-1266 cited in the application the whole document	1-3

INTERNATIONALER RECHERCHENBERICHT

Internationales Aktenzeichen

PCT/EP2007/051081

A. KLASSIFIZIERUNG DES ANMELDUNGSGEGENSTANDES
 INV. G06Q10/00

Nach der Internationalen Patentklassifikation (IPC) oder nach der nationalen Klassifikation und der IPC

B. RECHERCHIERTER GEBIETE

Recherchiertes Mindestprüfstoff (Klassifikationssystem und Klassifikationssymbole)
 G06Q

Recherchierte, aber nicht zum Mindestprüfstoff gehörende Veröffentlichungen, soweit diese unter die recherchierten Gebiete fallen

Während der internationalen Recherche konsultierte elektronische Datenbank (Name der Datenbank und evtl. verwendete Suchbegriffe)

EPO-Internal

C. ALS WESENTLICH ANGESEHENE UNTERLAGEN

Kategorie*	Bezeichnung der Veröffentlichung, soweit erforderlich unter Angabe der in Betracht kommenden Teile	Betr. Anspruch Nr.
X	OLKHOVICH L: "Semi-Automatic Business Process Performance Optimization Based On Redundant Control Flow Detection" TELECOMMUNICATIONS, 2005. AICT-ICIW '06. INTERNATIONAL CONFERENCE ON INTERNET AND WEB APPLICATIONS AND SERVICES/ADVANCED INTERNATIONAL CONFERENCE ON GUADELOPE, FRENCH CARIBBEAN 19-25 FEB. 2006, PISCATAWAY, NJ, USA, IEEE, 19. Februar 2006 (2006-02-19), Seiten 146-146, XP010898469 ISBN: 0-7695-2522-9 das ganze Dokument ----- -/--	1-3

Weitere Veröffentlichungen sind der Fortsetzung von Feld C zu entnehmen Siehe Anhang Patentfamilie

* Besondere Kategorien von angegebenen Veröffentlichungen :

"A" Veröffentlichung, die den allgemeinen Stand der Technik definiert, aber nicht als besonders bedeutsam anzusehen ist

"E" älteres Dokument, das jedoch erst am oder nach dem internationalen Anmeldedatum veröffentlicht worden ist

"L" Veröffentlichung, die geeignet ist, einen Prioritätsanspruch zweifelhaft erscheinen zu lassen, oder durch die das Veröffentlichungsdatum einer anderen im Recherchenbericht genannten Veröffentlichung belegt werden soll oder die aus einem anderen besonderen Grund angegeben ist (wie ausgeführt)

"O" Veröffentlichung, die sich auf eine mündliche Offenbarung, eine Benutzung, eine Ausstellung oder andere Maßnahmen bezieht

"P" Veröffentlichung, die vor dem internationalen Anmeldedatum, aber nach dem beanspruchten Prioritätsdatum veröffentlicht worden ist

"T" Spätere Veröffentlichung, die nach dem internationalen Anmeldedatum oder dem Prioritätsdatum veröffentlicht worden ist und mit der Anmeldung nicht kollidiert, sondern nur zum Verständnis des der Erfindung zugrundeliegenden Prinzips oder der ihr zugrundeliegenden Theorie angegeben ist

"X" Veröffentlichung von besonderer Bedeutung; die beanspruchte Erfindung kann allein aufgrund dieser Veröffentlichung nicht als neu oder auf erfinderischer Tätigkeit beruhend betrachtet werden

"Y" Veröffentlichung von besonderer Bedeutung; die beanspruchte Erfindung kann nicht als auf erfinderischer Tätigkeit beruhend betrachtet werden, wenn die Veröffentlichung mit einer oder mehreren anderen Veröffentlichungen dieser Kategorie in Verbindung gebracht wird und diese Verbindung für einen Fachmann naheliegend ist

"&" Veröffentlichung, die Mitglied derselben Patentfamilie ist

Datum des Abschlusses der internationalen Recherche	Absenddatum des internationalen Recherchenberichts
24. Mai 2007	12/06/2007

Name und Postanschrift der internationalen Recherchenbehörde Europäisches Patentamt, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 851 epo nl, Fax: (+31-70) 340-3016	Bevollmächtigter Bediensteter Streit, Stefan
---	---

C. (Fortsetzung) ALS WESENTLICH ANGESEHENE UNTERLAGEN

Kategorie*	Bezeichnung der Veröffentlichung, soweit erforderlich unter Angabe der In Betracht kommenden Teile	Betr. Anspruch Nr.
X	SEBASTIAN HINZ ET AL: "Transforming BPEL to Petri Nets. In: BPM 2005." LECTURE NOTES IN COMPUTER SCIENCE, Bd. 3649, 2005, Seiten 220-235, XP019017192 das ganze Dokument	1-3
A	AALST VAN DER W M P: "THE APPLICATION OF PETRI NETS TO WORKFLOW MANAGEMENT" JOURNAL OF CIRCUITS, SYSTEMS AND COMPUTERS, WORLD SCIENTIFIC, LONDON, GB, Bd. 8, Nr. 1, Februar 1998 (1998-02), Seiten 21-66, XP008038460 ISSN: 0218-1266 in der Anmeldung erwähnt das ganze Dokument	1-3