

(19)



Europäisches Patentamt

European Patent Office

Office européen des brevets



(11)

EP 1 265 751 B1

(12)

EUROPÄISCHE PATENTSCHRIFT

(45) Veröffentlichungstag und Bekanntmachung des
Hinweises auf die Patenterteilung:

15.10.2003 Patentblatt 2003/42

(51) Int Cl.7: **G06F 17/30, B41C 1/00**

(86) Internationale Anmeldenummer:

PCT/DE01/01051

(21) Anmeldenummer: **01923521.7**

(87) Internationale Veröffentlichungsnummer:

WO 01/068368 (20.09.2001 Gazette 2001/38)

(22) Anmeldetag: **19.03.2001**

(54) **VERFAHREN UND VORRICHTUNG ZUR ERSTELLUNG EINER DRUCKVORLAGE**

METHOD AND DEVICE FOR PRODUCING A PRINTING PATTERN

PROCEDE ET DISPOSITIF POUR REALISER UNE MAQUETTE POUR IMPRESSION

(84) Benannte Vertragsstaaten:

**AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU
MC NL PT SE TR**

• **WEIHER, Marcel Peter**

14055 Berlin (DE)

• **ZÜHLSDORF, Ralph**

14532 Kleinmachnow (DE)

(30) Priorität: **17.03.2000 DE 10013096**

(43) Veröffentlichungstag der Anmeldung:

18.12.2002 Patentblatt 2002/51

(74) Vertreter: **Rumpler, Wolfgang, Dr. et al**

Hofstetter, Schurack & Skora,

Patentanwälte,

Balanstrasse 57

D-81541 München (DE)

(73) Patentinhaber: **Cidentigo GmbH**

10553 Berlin (DE)

(56) Entgegenhaltungen:

EP-A- 0 837 401

US-A- 5 615 316

US-A- 5 740 338

(72) Erfinder:

• **SCHMIDT-JOOS, Herwart**

12157 Berlin (DE)

Anmerkung: Innerhalb von neun Monaten nach der Bekanntmachung des Hinweises auf die Erteilung des europäischen Patents kann jedermann beim Europäischen Patentamt gegen das erteilte europäische Patent Einspruch einlegen. Der Einspruch ist schriftlich einzureichen und zu begründen. Er gilt erst als eingelegt, wenn die Einspruchsgebühr entrichtet worden ist. (Art. 99(1) Europäisches Patentübereinkommen).

EP 1 265 751 B1

Beschreibung

GEBIET DER ERFINDUNG

[0001] Die vorliegende Erfindung betrifft ein Verfahren zur Bearbeitung und Erstellung von elektronischen Publikationsdokumenten und die geeignete Vorrichtung hierzu. Bei den Publikationsdokumenten gibt es keine Einschränkung für deren späteren Gebrauchszweck. Insbesondere handelt es sich um Druckvorlagen wie z. B. Anzeigen aus dem Zeitungs- und Zeitschriftenbereich, aber auch Publikationsdokumente für die Darstellung auf beliebigen elektronischen, visuellen Anzeigegeräten.

BESCHREIBUNG DES STANDES DER TECHNIK

[0002] Der respektive Prozess, der sich bei Publikationsdokumenten durch Inhaltsänderungen auf das Layout auswirkt ist aufwendig und damit kosten- und zeitintensiv. Außerdem sind *sehr* häufig Qualitätsverluste durch unbeabsichtigte Layoutveränderungen oder gar der Zerstörung der Gestaltung mit jeder Inhaltsänderungen verbunden.

[0003] Selbst einfache inhaltliche Änderungen können nur von Experten wie Grafiker, Reinzeichner oder DTP-Profis vorgenommen werden, die auf komplizierte Grafikprogramme geschult sein müssen. Dieser Umstand bindet Kreativitätspotential und verschwendet Arbeitszeit für unnötige automatisierbaren Aufgaben.

[0004] Bei Inhaltlichen Änderungen in Publikationsdokumenten durch Laien besteht gegenwärtig die Gefahr der Zerstörung des einheitlichen Corporate Designs. Der Hauptgrund hierfür besteht darin, dass heutige Publikationsdokumente keine Intelligenz enthalten, die eine Zerstörung der zugrundeliegenden Gestaltung bei manuellem Eingriff ausschließen.

[0005] US-A-5 740 338 beschreibt ein Verfahren, bei dem gerastete Bilddaten mit einer Bildvorlage zusammengefügt und anschließend auf einem Drucker ausgegeben werden. Somit entsteht ein neues Bild, aber keine neue Vorlage, die selbst variabel wäre.

EP-A-0 837 401 beschreibt ein Verfahren, bei dem Vorlagen bereitgestellt werden, welche Platzhalter in Dokumentpositionen als austauschbare Teile enthält, die von einem Prozess mit unterschiedlichen Daten versorgt werden. Den Platzhaltern sind Variablen zugeordnet und es soll primär ermöglicht werden, dass gleicher Inhalt in verschiedenen Layouts auf unterschiedlichen Druckern ausgegeben werden kann. Auch hier handelt es sich um das Erstellen einer finalen Druckvorlage und nicht um das Erstellen einer variablen Druckvorlage.

[0006] Die Aufgabe der Erfindung besteht darin, die beschriebenen Nachteile im Stand der Technik zu vermeiden und ein Verfahren, eine Vorrichtung, sowie ein "intelligentes" Dokumentenformat und ein Computerprogramm zur ferngesteuerten Erstellung und Bearbeitung von Publikationsdokumenten vorzuschlagen. Neben

der Trennung von Inhalt und Design wird bei Inhaltsanpassungen, über die in diesen intelligenten Publikationsdokumenten enthaltenen Regeln sichergestellt, dass keine ungewollten Designänderungen entstehen können.

[0007] Der ökonomische Erfindungszweck besteht in dem kosten- und zeitsparenden, wirtschaftlichen Nutzen gegenüber den mit heutigem Stand der Technik vergleichbaren Abläufen. Beim konsequenten Einsatz der Erfindung ist zu erwarten, daß sich der Produktionsablauf bei der klassischen Publikationserstellung stark vereinfachen wird und neue effektivere Arbeitsbedingungen schaffen wird.

15 ZUSAMMENFASSUNG DER ERFINDUNG

[0008] Erfindungsgemäß wird die Aufgabe gelöst durch ein Verfahren nach Anspruch 1 und eine Vorrichtung nach Anspruch 10. Vorteilhafte Weiterbildungen ergeben sich aus den Unteransprüchen.

[0009] Die Erfindung kann als ein Softwaresystem zur Steuerung der zugrundeliegenden Templatetechnologie realisiert werden, wobei das Softwaresystem aufweist: eine Einrichtung zur Erstellung und Bearbeitung der Templates auf einem entfernten Computersystem (Server), wobei die Template-Erstellung/Bearbeitung auf dem Server stattfindet und ferngesteuert über einen lokalen Webbrowser, wie er heute standardgemäß im Internet Verwendung findet bedient wird; eine Einrichtung zur Erstellung und Bearbeitung der Templates auf einem lokalen Computer ist eine zusätzliche Möglichkeit diese Templates auch ohne Server zu erstellen und bearbeiten; eine Einrichtung die es ermöglicht ferngesteuert Inhaltsänderungen in Publikationsdokumenten auf einem Server-Computersystem vorzunehmen und anschließend Design und Inhalt in den Templates neu zusammenzuführen, wobei der ferngesteuerte Eingabedialog zum einen über einen Standard-Web-Browser erfolgen kann, dem zur Eingabekontrolle eine Voransicht des geänderten Publikationsdokumentes dem jeweils initiiierenden Browserdialog bereitgestellt wird und zum anderen über sogenannte Batch-Schnittstellen zu Fremdprozessen automatisch mit neuen Inhalten versorgt wird.

[0010] Die Erfindung erlaubt zum Beispiel die schwierige Aufgabe, im Rahmen der Corporate Design Richtlinien einer Marke bzw. einer Kampagne eine ansprechende Anzeige zu erstellen, für den Endanwender auf die sehr viel einfachere Aufgabe reduziert, zwischen Varianten auszuwählen. Die Gesamtaufgabe ist damit nicht leichter geworden, erlaubt es aber, den komplexen Teil einmal für alle nachfolgenden Prozesse zu erledigen. Darüber, wie genau die Aufteilung beschaffen ist, entscheidet nicht das Kernel, sondern die mit dem Kernel verwendeten "intelligenten" Templates.

[0011] Die Erfindung kann beispielsweise dazu verwendet werden, dass lokale Werbetreibende (z. B. Handelspartner, Niederlassungen, beliebige Personen) mit

zentral vorgegebene Gestaltungsvorlagen unter Wahrung eines vorgegebenen Qualitätsstandards über das Internet Inhaltsänderungen vornehmen können ohne dabei das Design zu zerstören.

Anwendungsfall am Beispiel "Anzeigenlokalisierung"

[0012] Eine vom Markeninhaber beauftragte Agentur erstellt z. B. eine Anzeige, die seinem festgelegten Corporate Design entspricht. Diese Anzeige dient als Musteranzeige. Die Erfindung ermöglicht dem lokalen Werbetreibenden über das Internet die Modifizierung oder den Austausch einzelner "genehmigter" Text- und Bildelemente unter Beibehaltung aller für das Corporate Design relevanten Merkmale. Die Anpassung der Musteranzeige kann von den lokalen Werbetreibenden selbstständig in einem bedienerefreundlichen Browserfenster vorgenommen werden. Die Einbeziehung regionaler Agenturen entfällt.

[0013] Die zentral erstellte Musteranzeige und die anzupassenden Anzeigenelemente werden zur weiteren Bearbeitung in ein Computerprogramm eingelesen. Diese Musteranzeige und ihre Ersetzungselemente werden z. B. von Layoutexperten einer Werbeagentur mit Standard-Publishing-Programmen produziert und z. B. als EPS-Datei zur Verfügung gestellt. Damit ist eine problemlose Anpassung an die bestehende Software-Umgebung gesichert. Sowohl die Anzeigenvorlage als auch die Anzeigenelemente, die den Vertragspartnern für die Modifizierung ihrer Lokalanzeige zur Verfügung stehen, können auf gewohnte Weise erstellt werden.

[0014] Der Markeninhaber und die beauftragte Agentur legen mit den Corporate Design-Vorgaben zentral fest, welche Anzeigenelemente fest, welche Anzeigenelemente Variabel sein sollen. Außerdem werden die Ersetzungselemente definiert.

[0015] Die Anpassung der Anzeige nehmen die Vertragspartner in einem Browserfenster vor. Ein Computerprogramm auf einem entfernten Computer (Server), der über das Internet mit dem Computer (Client) des Vertragspartners verbunden ist realisiert die Integration der modifizierten Anzeigenelemente unter genauer Berücksichtigung des festgelegten Corporate Design.

[0016] Als Ergebnis entstehen einheitliche Anzeigen, die alle signifikanten Merkmale des vorgegebenen Corporate Designs aufweisen und die angepassten Elemente der jeweiligen Vertragspartner enthalten. Das zentrale Marketing-Management erfährt eine deutliche Effizienzsteigerung, da durch dieses Verfahren die Corporate Design-Vorgaben systematisch gesichert sind. Abstimmungen mit lokalen Agenturen entfallen.

Template Technologie

[0017] Die Template-Technologie ermöglicht durch parametrisierte Druckvorlagen die Trennung von Design und Inhalt und damit erhebliche Effizienzsteigerun-

gen bei der Verarbeitung von Design in Publishingdokumenten. Design und Inhalt können jeweils getrennt erstellt und wiederverwendet werden. Die bisher übliche Wiederholung von aufwendigen, kostenintensiven Prozessen und fehleranfälligen manuellen Arbeitsschritten bei inhaltlichen Änderungen entfällt.

[0018] Die Template Technologie basiert auf grundlegenden Mechanismen der Informatik, die auf eine neue Art und Weise kombiniert werden. Zu diesen Mechanismen gehören Variablen und die Bindung von Variablen an Werte, partielle Auswertung, formale Parameter etc.

[0019] Es werden Template-Objekte definiert. Wie normale Objekte in der objektorientierten Programmierung haben diese Template-Objekte einen Zustand der durch die Belegung ihrer Variablen definiert wird und "Verhalten" das durch diese Belegung beeinflusst wird.

[0020] Im Gegensatz zu normalen Objekten können in Template-Objekten Variablen aber auch (noch) unbelegt sein [Template-Currying], sie verhalten sich damit wie die Parameter von Prozeduren. Die unbelegten (freien) Variablen sind Parameter des Templates. Ein Template-Objekte "weiss" welche seiner Parameter frei und welche gebunden sind. Nur freie Parameter können extern beeinflusst werden.

[0021] Für maximale Flexibilität sollten möglichst viele [alle] Parameter frei gehalten werden. Das Template-Currying sowie Defaults erlauben dies ohne die Komplexität für den Endanwender unmöglich werden zu lassen.

[0022] Ein Template-Objekt wird erst durch die Kombination mit einer Belegung aller seiner freien Variablen vollständig. Eine Belegung ist eine Zuordnung von Variablen(-namen) zu Werten. Durch sie werden die freien Parametervariablen eines Templates an spezifische Werte gebunden. Sind sie einmal gebunden stehen sie nicht mehr weiter als Parameter zur Verfügung.

[0023] Template-Objekte können auch mit unvollständigen Belegungen kombiniert werden, also solchen in denen nicht alle freien Variablen des Template-Objekts an einen Wert gebunden werden. Aus dieser Kombination von Template+Bindungen entsteht ein neues Template-Objekt mit den jetzt noch verbleibenden freien Variablen als Parameter. Dieses neue Objekt bezieht sich dabei weiter auf seine Konstituenten.

[0024] Durch diesen Mechanismus können Templates exakt auf bestimmte Anwendungen zugeschnitten werden: der Template-Konfigurator belegt alle Parameter bis auf die, die vom Endanwender veränderbar sein sollen. [Dieser Punkt ist von zentraler Bedeutung]

[0025] Es können sogenannte Default-Belegungen definiert werden. Diese binden wie normale Belegungen Werte an Parametervariablen, die so gebundenen Variablen stehen jedoch weiterhin für Bindungen zu Verfügung. Neuere Bindungen haben dabei Priorität über alte Bindungen.

[0026] Bindungen können als Werte auch wieder Templates enthalten, diese Templates sind dann "Unter-Templates" des Templates an das sie gebunden wer-

den. Dadurch kann eine Hierarchie von Templates aufgebaut werden.

[0027] Die ungebundenen Parameter aller untergeordneten Templates werden dem sie einbindenden Template zugeordnet. Die möglicherweise komplexe Unterstruktur eines Templates bleibt somit im Normalfall versteckt, kann aber von entsprechenden Werkzeugen sichtbar gemacht werden. (Werkzeuge für Template-Ersteller). [Abstraktion]

[0028] Anstatt eines spezifischen Werts kann eine Variable auch an eine andere Variable gebunden werden. Damit ist ihr Wert abhängig vom Wert dieser anderen Variable.

[0029] Eine besondere Art der abhängigen Bindung ist die strukturelle Vererbung. Dabei wird der Wert einer Parametervariable nicht an eine spezifische andere Variable gebunden, sondern es wird der Wert einer Variable gleichen Namens eines übergeordneten Templates übernommen.

[0030] Bei struktureller Vererbung kann es passieren, dass eine freie Variable durch diesen Mechanismus keinen Wert zugewiesen bekommt. Es sollte daher zuvor ein Default-Wert definiert werden.

[0031] Strukturelle Vererbung kann auch anders definiert werden: als abhängige Bindung mit mehreren Outputs (also push vs. pull). Dadurch werden undefinierte Zustände vermieden.

Der Wert einer Variable kann auch an eine beliebige Prozedur gebunden werden, die diesen Wert berechnet. (Dies ergibt sich aus (a) der Tatsache das Prozeduren ein Spezialfall von Templates sind und (b) Werte auch Templates sein können. Weiterhin ergibt sich, dass eventuell Parameter der Prozedur sich wieder in die Template/Bindungs Systematik einfügen).

[0032] Es können auch Beziehungen zwischen Variablen definiert werden. Mit diesen lassen sich gültige Variablenbindungen vom System errechnen oder es wird festgestellt, dass es keine gültige Belegung gibt. Diese Beziehungen können z. B. arithmetisch, logisch sein, Ungleichheiten enthalten etc. Abhängige Bindungen sind z. B. nur ein Spezialfall von Beziehungen.

[0033] Es können aus Templates durch Verfeinerung neue Templates mit zusätzlichem Verhalten und zusätzlichen freien Variablen (und/oder Bindung bestimmter vorhandener freier Variablen) erzeugt werden.

[0034] Aus konkreten Objekten können (wieder) Templates erstellt werden indem feste Attribute/Objekte durch eine Parametervariable ersetzt werden (dabei wird der derzeitige Wert üblicherweise als Default übernommen). Dies ist möglich sowohl für Unterobjekte als auch für Attribute.

[0035] Damit Daten aus unstrukturierten Dateiformaten in das Template-Konzept einbezogen werden können, müssen automatisch oder assistiert Strukturen erkannt werden.

Kernel

[0036] Der Kernel erlaubt es, zusammen mit der dazugehörigen Template-Technologie, die Komplexität von Problemstellungen bei der Erstellung von Publikationsdokumenten aufzuteilen und Teile getrennt voneinander zu behandeln. Von großer Bedeutung ist hier die Trennung in die Designobjekte und deren Bearbeitung. Die Designobjekte sind vollständig durch ihre jeweiligen Templates beschrieben und können über den Kernel bearbeitet bzw. ausgewertet werden.

[0037] Die Templates stehen für die unterschiedlichen Designvarianten zur gesteuerten Variantenmodifikation zur Verfügung. Über die beschriebene Parameterbindung wird nur auf die zulässigen "gewollten" Gestaltungsvariationen eingeschränkt. Als Zusatzinformationen können weitere Gestaltungsregeln automatisch "gewollte" Designvarianten erzeugen. Alle Designvarianten liegen in Form von Templates als elektronische Repräsentation auf einem digitalen Speichermedium und können über den Kernel verarbeitet, variiert, dupliziert und als direkte elektronische Vorlage für nachgeschaltete Prozesse umgewandelt werden. Geeignete Formate für nachgeschaltete Prozesse sind derzeit z. B. EPS, TIFF, PDF, SVG, HTML und XML, es gibt jedoch keine Beschränkung auf diese beispielhafte Auswahl.

[0038] Der Kernel stellt eine stets erweiterbare virtuelle Maschine zur Verarbeitung von Templates zu Templatevarianten also abgeleiteten Templates dar. Erweiterbar bedeutet hier, dass der Kernel den Anforderungen entsprechend in seiner Mächtigkeit beliebig wachsen kann. Nur der Kernel enthält das Wissen darüber wie die Templates und die darin enthaltenen Regeln auszuführen bzw. umzusetzen sind. Er ist also die zentrale Vorrichtung für die Erstellung und Bearbeitung von Templates.

Verfahren zur Erstellung von Templates und Templatevarianten

[0039] Das Verfahren zur Erstellung von Templates basiert vorzugsweise auf einem Computerprogramm und kann sowohl auf einem lokalen als auch einem entfernten Rechner ausgeführt werden. Die Ausführung kann durch eine natürliche Person oder durch ein weiteres Computerprogramm ferngesteuert werden. Das ferngesteuerte Verfahren der Erstellung und Bearbeitung bedient sich hierbei derzeit Standardtechnologien wie z. B. Internet, Intranet und World Wide Web oder WAP, beschränkt sich allerdings nicht hierauf.

[0040] Die Erstellung von Templates geschieht in folgenden Phasen:

(a) Einlesen logisch unstrukturierter aber graphisch strukturierter Druckvorlagen in druckbereiten Formaten

(b) logisches Erkennen und Analysieren von graphischen Strukturmerkmalen innerhalb der Druck-

vorlage

(c) Erzeugen einer logischen Komponentenstruktur aus den graphischen Strukturmerkmalen

(d) Zuordnung dieser logischen Komponenten zu beliebig parametrisierbaren Substitutionselementen (beliebig parametrisierbar = maximale Freiheit)

(e) Vereinfachung der Substitutionselemente durch Festlegung von Parametern

- Aufteilung der Parameter in feste und variable Anteile
- Aufteilung wird vom Benutzer festgelegt, nicht von der Programmstruktur

(f) Erzeugung einer parametrisierten Vorlage (Template) aus:

- der Originalvorlage ihrer Komponentenstruktur (via Strukturmerkmale)
- den Substitutionselementen ihrer festen und variablen Parametern
- den Vorgaben für die variablen Parameter
- der Zugang zwischen Substitutionselementen und Komponenten der Druckvorlage

[0041] Verfahren zur Erzeugung von Gestaltungsvarianten aus Templates

[0042] Das Verfahren zur Erzeugung von Gestaltungsvarianten basiert vorzugsweise ebenfalls auf einem Computerprogramm, das auf einem lokalen und einem entfernten Rechner ausgeführt werden kann. Die Ausführung kann durch eine natürliche Person oder durch ein weiteres Computerprogramm ferngesteuert werden.

[0043] Die Erzeugung von Gestaltungsvarianten geschieht in folgenden Phasen:

(g) Automatische Erzeugung von Formularen aus der parametrisierten Druckvorlage über die Auswertung der Templates

(h) Automatische Erzeugung einer Voransicht aus der Druckvorlage mit den aktuellen Werten der variablen Parameter (optional)

(i) Substitution der variablen Parameter der parametrisierten Druckvorlage durch Ausfüllen des Formulars oder durch von einem Computerprogramm erzeugten Werten

(j) Anpassung der Druckvorlage an die aktuelle Parameterlage der Substitutionselemente

(k) Ablage der so erzeugten Gestaltungsvariante als Template mit enthaltener parametrisierter druckfähiger Vorlage auf einem digitalen Speichermedium wie z. B. Festplatte CD-ROM, DVD etc.

KURZBESCHREIBUNG DER ZEICHNUNGEN

[0044] Durch die folgenden Ausführungen in Verbindung mit den Zeichnungen werden die Bedeutung und

die Vorteile der Erfindung weiter verdeutlicht, in denen

Figur 1 ein Flußdiagramm zur Erläuterung der Phasen zur Erstellung eines Templates gemäß einem Ausführungsbeispiel der vorliegenden Erfindung zeigt;

Figur 2 ein Fußdiagramm zur Erläuterung der Verfahrensschritte zur Erstellung einer Gestaltungsvariante über das Internet gemäß einem Ausführungsbeispiel der vorliegenden Erfindung zeigt;

Figur 3 eine schematische Darstellung eines Templates gemäß einem Ausführungsbeispiel der vorliegenden Erfindung zeigt;

Figur 4 eine schematische Darstellung eines fertiggestellten Gestaltungsvariante als parametrisierte druckfähige Vorlage zeigt; und

Figur 5 eine schematische Darstellung einer bevorzugten erfindungsgemäßen Hardware-Konfiguration ist.

25 DETAILIERTE BESCHREIBUNG EINES AUSFÜHRUNGSBEISPIELS

[0045] Die Erfindung wird im folgenden unter Bezugnahme auf die Zeichnungen im Detail erläutert. Zunächst seien die verwendeten Begriffe näher definiert:

- Eine logisch **unstrukturierte Druckvorlage** bezeichnet ein in elektronischer Form auf einem Speichermedium gehaltenes Dokument, das mit einem Standard-Publishing-Programm erstellt und graphisch strukturiert wurde. Die unstrukturierte Druckvorlage enthält alle elektronisch kodierten Informationen die für die Visualisierung des zugrundeliegenden Dokuments erforderlich ist. Es gibt keine systematische Aufteilung logischer Visualisierungs-Komponenten, wie z. B. Textspalten, Überschriften, Bilder, Logos etc., aus denen sich das Dokument zusammensetzt. Vielmehr werden bei der Darstellung unstrukturierter Druckvorlagen die Pixel einzeln angesteuert.

- Aus graphischen **Strukturmerkmalen** werden logische Visualisierungskomponenten von kodierten Visualisierungsverfahren über ihren zugrundeliegenden Programmcode abgeleitet.

- Eine **Komponentenstruktur** bezeichnet eine systematische Unterteilung der logischen Visualisierungs-Komponenten, wie z. B. Textspalten, Überschriften, Bilder, Logos etc., aus denen sich das Dokument zusammensetzt.

- **Parametrisierbare Substitutionselemente** be-

zeichnen logischen Visualisierungs-Komponenten, wie z. B. Textspalten, Überschriften, Bilder, Logos etc., die durch freie oder eingeschränkte Parameterwertebereiche durch z. B. externen, programm-gesteuerten Eingriff in Inhalt und Form verändert werden können.

- Ein **parametrisierte Vorlage (Template)** bezeichnet eine strukturierte Vorlage, die in elektronischer Form auf einem Speichermedium gehalten, zusätzlich zur druckfähigen Vorlage ihre bezugnehmende systematische Aufteilung der logischen Visualisierungs-Komponenten, wie z. B. Textspalten, Überschriften, Bilder, Logos etc. enthält. Die Druckfähige Vorlage wird ebenfalls in elektronischer Form in einem bekannten Standardformat wie z. B. EPS, TIFF, PDF, SVG, HTML und XML, auf einem Speichermedium gehalten. Der Ablageort der druckfähigen Vorlage muss dabei nicht notwendigerweise mit dem Speicherort des zugehörigen Templates übereinstimmen. Es ist ausreichend, wenn sich im Template eine Referenz befindet, wie z. B. eine im Internet übliche URL, über die der genaue Speicherort über das Internet lokalisiert werden kann.
- Ein **Gestaltungsvariante** bezeichnet eine parametrisierte Vorlage (Template), welche z. B. durch externen programmgesteuerten Eingriff in Inhalt und Form verändert wurde und alle Komponenten, einschließlich der druckfähigen Vorlage, in einem Datenpaket (Bundle) auf einem elektronischen Speicher hält.

Jede fertiggestellte Gestaltungsvariante bezeichnet ein durch die Erfindung auf der Basis der unstrukturierten Vorlage abgeleitetes Template, wobei aus jedem Template beliebig viele Gestaltungsvarianten erzeugt werden können. Dieses liegt vorzugsweise in elektronischer Form vor und dient z. B. als Werbemittel, wie etwa Anzeigen aller Art, Broschüren, Zeitschriften aber auch sogenannte Akzidenzdrucksachen wie: Biebfögen, Visitenkarten, Gratulationskarten, Trauerkarten. Neben den klassischen Drucksachen erhalten zunehmend personalisierte Publikations-Dokumente Bedeutung. Diese eignen sich besonders für die direkt auf elektronischem Wege erreichbare Ausgabe auf Digitaldruckern, CTP-Systemen und Bildschirmen über das Internet.

Eine fertiggestellte Gestaltungsvariante kann als Shop-Produkt and den Besteller geliefert werden. Ein Shop-Produkt kann zusätzlich weitere Informationen wie z. B. Preisangaben für bestimmte Mengen enthalten.

[0046] Figur 1 zeigt schematisch die wesentlichen Schritte zur Erstellung eines erfindungsgemäßen Templates durch den Designer oder Template-Ersteller. Im ersten Schritt 11 wird eine logisch unstrukturierte Druck-

vorlage in das erfindungsgemäße Computerprogramm MetaAd Creator zur Erstellung von Templates eingele-
sen. Diese unstrukturierte Druckvorlage wurde z. B. mit
einem Standardprogramm erstellt und befindet sich in
einem druckbereiten Format wie z. B. EPS, PS, PDF,
SVG etc. und wird dann in Verfahrensschritt 12 durch
das erfindungsgemäße Computerprogramm MetaAd
Creator analysiert und auf graphische Strukturmerkmale untersucht. In 13 wird aus den gefundenen wesentlichen
Strukturmerkmalen eine logische Komponenten-
struktur aufgebaut. Die Komponentenstruktur umfasst
gegebenenfalls feste und/oder variable logische Komponenten. Zu den variablen Komponenten passende
parametrisierbare Substitutionselemente werden dem
Designer oder Template-Ersteller als gebundene Parameternamen über das Benutzerinterface im MetaAd
Creator angezeigt In Schritt 14 erfolgt die Zuordnung
der jeweils interessierenden variablen Komponenten zu
beliebig parametrisierbaren Substitutionselementen.
Die Parameter können durch den Designer oder Template-Ersteller in ihrem Wertebereich auf einen Wertevorrat eingeschränkt werden. Es kann z. B. eine bestimmte Farbpalette oder eine festgelegte Gruppe von
Werbetexten als Substitutionselemente zugeordnet
werden. Manchmal ist es jedoch sinnvoll den Wertevorrat nicht einzuschränken. Ein Beispiel hierfür ist, wenn
künftige Benutzer die Möglichkeit erhalten sollen Textkomponenten durch beliebige eigene Texte ersetzen zu
können. In Schritt 15 wird die parametrisierte Druck-
oder Designvorlage (Template) erzeugt. die alle variablen bzw. Ersetzungskomponenten und/oder Regeln
enthält. Diese Designvorlage wird nach der Speicherung auf einem digitalen Datenträger wie z. B. Festplatte
als MetaAd Template bezeichnet und steht sofort als
Vorlage zur Erzeugung einer beliebigen Anzahl von Gestaltungs- oder Designvarianten zur Verfügung. Es kann
auch ein geeignetes Archivsystem über eine beliebig konfigurierbare Schnittstelle zur Archivierung programmgesteuert angeschlossen werden. Hierin könnten
MetaAd Templates, Gestaltungsvarianten und auch Substitutionselemente archiviert und verwaltet werden.
[0047] Figur 2 zeigt schematisch die wesentlichen Schritte zur Erstellung einer erfindungsgemäßen Designvariante durch den Besteller oder Benutzer. Im ersten
Schritt 21 wird die Einwahl des Bestellers oder Benutzers vollzogen. Anschließend wird in 22 seine Autorisierung durch Überprüfung eines individuellen Kennwortes durch das erfindungsgemäße Computerprogramm MetaAd Web-Deploy vorgenommen. Nach Bestätigung
seiner Zugangsberechtigung wird ihm die ferngesteuerte Bearbeitung seiner MetaAd Templates durch MetaAd Web-Deploy ermöglicht. In Schritt 23 wird eine Designvariante bzw. Template vom Besteller oder Benutzer
ausgewählt In Schritt 24 werden durch Zuordnung der Substitutionselemente zu den variablen Parametern
bzw. Komponenten Designvarianten der MetaAd Templates erstellt. Dabei bezeichnet Designvarianten alle
die MetaAd Templates, welche auf diese Weise aus dem

ursprünglich vom Designer oder Template-Ersteller erstellten Original-Template, entstanden sind.

[0048] Das Original-Template ist strukturgleich, wird jedoch nicht überschrieben durch die vorgenommenen Änderungen. Bei den Substitutionselementen können z. B. Bilder oder Textelemente eines fest vorgegebenen Wertebereichs durch Besteller- oder Benutzer-Auswahl ersetzt werden. Der Wertebereich für Substitutionselemente muss aber nicht notwendig eingeschränkt sein sondern kann auch durch den Ersteller oder Designer zur uneingeschränkten Änderung von Parameterwerten geöffnet sein. Zum Beispiel kann dies sinnvoll sein, wenn für Texte die freie Texteingabe oder die Zuordnung von Farben zu Texten oder Geometrien ermöglicht werden soll.

[0049] Die Substitutionselemente können ferner als Ergebnisse von Programmaufrufen bereitgestellt werden. In diesem Fall wird das Modul MetaAd Batch-Deploy zur automatischen Parameterbindung durch programmgenerierte Werte benutzt. Durch MetaAd Batch-Deploy kann auch die gesamte Besteller- oder Benutzer-Dialog entfallen und alle Designvarianten durch die programmgesteuerte Bereitstellung von Substitutionselementen bestimmt werden.

[0050] In Schritt 25 werden die im MetaAd Template eingebetteten Druckvorlagen an die aktuelle Parameterlage der Substitutionselemente angepasst. Hiermit ist die temporäre Designvariante fertiggestellt. Bei Bedarf wird in Schritt 26 ein Vorschaubild zur Überprüfung erzeugt, welches für die Visualisierung der Designvariante zum Web-Browser des Bestellers oder: Benutzers gesendet wird. Hierdurch wird ein interaktive Arbeitsweise mit unmittelbarer Kontrolle durch die Visualisierung aller vorgenommenen Veränderungen von Inhalt und Gestaltung in einer Designvariante oder des jeweiligen MetaAd Original-Templates ermöglicht.

[0051] In Schritt 27 wird diese Designvariante zur Archivierung im Besteller- oder Benutzerverzeichnis für die einen Designvarianten abgelegt. Es kann auch ein geeignetes Archivsystem über eine beliebig konfigurierbare Schnittstelle zur Archivierung programmgesteuert angeschlossen werden. Hierin könnten MetaAd Templates, Gestaltungsvarianten und auch Substitutionselemente archiviert und verwaltet werden.

[0052] In Schritt 28 wird optional ein Bestellvorgang eingeleitet. Der Besteller hat hier die Möglichkeit eine Warenkorbfunktion oder andere Kauf- und Bezahlvorgänge zu starten und einen Druckauftrag zu platzieren. Nach Eingabe der Druckauftragsdaten und Absenden der Auftragsbestätigung wird über das MetaAd Template ein Druckauftrag in einer Druckereinrichtung generiert. Die Druckauftragsdaten können ebenfalls im MetaAd Template enthalten sein. Die komplette Abwicklung des so platzierten Druckauftrages einschließlich Bezahlung kann über MetaAd Web-Deploy erfolgen.

[0053] In Schritt 29 wird die druckfähige Vorlage aus der Designvariante unter Berücksichtigung der Druckauftragsdaten gedruckt.

[0054] Figur 3 zeigt ein MetaAd Template mit festen bzw. statischen und variablen bzw. dynamischen Visualisierungs-Komponenten, wobei die dynamischen Visualisierungs-Komponenten im MetaAd Creator als Defaultwerte an Parameter gebunden werden, welche dann zu einem späteren Zeitpunkt durch Substitutionselemente ersetzt werden. Dieser Ersetzungsvorgang findet über die Anwendung der erfindungsgemäßen Programme MetaAd Web-Deploy, MetaAd Batch-Deploy, MetaAd GUI-Deploy oder aber erneut MetaAd Creator an dem MetaAd Template statt.

[0055] Bei 31 kann das Logol z. B. durch andere Logos ersetzt werden, die als Substitutionselemente in Frage kommen. Die statischen Visualisierungs-Komponenten 32 und 33 sind fest an Parameter gebunden und nicht für die spätere Ersetzung freigegeben.

[0056] Durch 34 und 37 soll angezeigt werden, dass über eine MetaAd-Anwendung eine Ersetzung durch beliebigen Text erfolgen kann. Die Schriftwahl und die Schriftgröße sind in diesem Beispiel fest vorgegeben und es kann nur eine Inhaltsänderung erfolgen. Allerdings könnte hier auch durch weitere freie Parameter zu diesem Substitutionselement z. B. die Schriftfarbe, Type, Größe etc. variabel gehalten werden.

[0057] Bei den Komponenten 35 und 36 handelt es sich um dynamische Visualisierungs-Komponenten vom Typ Bild. Auch hier sind den freien Parametern zunächst Defaultwerte zugeordnet, welche dann durch eine MetaAd-Anwendung ersetzt werden können.

[0058] Figur 4 zeigt eine Gestaltungsvariante, die strukturgleich zum MetaAd Original-Template aus Figur 3 ist. Wesentlich ist bei dieser Figur, dass sie eine konkrete Gestaltungsvariante darstellt, welche durch den Ersetzungsmechanismus einer MetaAd-Anwendung auf ein MetaAd Original-Template entstanden ist.

[0059] Das als Defaultwert gebundene Logol des Parameters aus 31 ist durch ein anderes Logo 41 ersetzt worden. Bei 42 und 43 konnte keine Änderung erfolgen. Durch 44 und 47 wird gezeigt, daß die freien Parameter aus Figur 3 durch neue Texte ersetzt wurden.

[0060] Figur 5 zeigt eine bevorzugte Hardware-Konfiguration zur Ausführung der Erfindung. Bezugszeichen 51 bezeichnet Besteller oder Benutzer, die mittels Client-Computer in einem Standard-Webbrowser über das Internet oder Intranet z. B. Designvarianten von vorliegenden Templates auf einem zentralen Server erzeugen. Die Verfahrensschritte zur Erzeugung von Designvarianten wurden unter Bezugnahme auf Figur 2 erläutert.

[0061] Bezugszeichen 52 bezeichnet einen oder mehrere Designer oder Template-Ersteller, die auf ihrem lokalen Computer Templates erstellen und über das Inter- oder Intranet auf einem zentralen Server übertragen. Sie können diese aber auch mittels Client-Computer in einem Standard-Webbrowser direkt über das Inter- oder Intranet erzeugen.

[0062] Bezugszeichen 53 bezeichnet die Verbindung zwischen Besteller, Benutzer, Designer oder Template-

Ersteller und Anwendungsserver über das Inter- oder Intranet.

[0063] Bezugszeichen 54 bezeichnet den Anwendungsserver, auf dem die Web-Applikation die Ausführung der Anweisungen der Besteller, Benutzer und Designer oder Template-Ersteller vollzieht, wie unter Bezugnahme auf Figur 2 erläutert wurde. Die Druckvorlagen, Substitutionselemente etc. werden in einer geeigneten Datenhaltung 55 bereitgestellt. Die Hardwarekonfiguration ist nicht auf einen bestimmten Computertyp festgelegt.

Patentansprüche

1. Verfahren zum Erstellen von variablen Druckvorlagen durch

Bereitstellen einer logisch unstrukturierten und graphisch strukturierten Basisdruckvorlage, welche in Form eines Programmcodes vorliegt,

Automatisches Erkennen (12) mindestens eines graphischen Strukturmerkmals (31 bis 37) der Basisdruckvorlage,

Automatisches Erzeugen (13) mindestens einer logischen Komponente auf der Grundlage des mindestens einen graphischen Strukturmerkmals (31 bis 37) und

Automatisches Erstellen (15) einer variablen Druckvorlage unter Verwendung der mindestens einen logischen Komponente.

2. Verfahren nach Anspruch 1, wobei das mindestens eine graphische Strukturmerkmal (31 bis 37) die graphische Darstellung einer logischen Komponente wie die eines Wortes, einer Überschrift, einer Textspalte, eines Bildes oder dergleichen ist.

3. Verfahren nach Anspruch 2, wobei die logische Komponente durch ein Substitutionselement, das individuell erzeugt oder aus einer Vielzahl auswählbar ist, ersetzt wird.

4. Verfahren nach Anspruch 3, wobei das Substitutionselement selbst eine parametrisierte oder nicht parametrisierte Druckvorlage ist

5. Verfahren nach einem der Ansprüche 1 bis 4, wobei die mindestens eine logische Komponente parametrisiert wird, indem die dem mindestens einen Strukturmerkmal zugrunde liegende Graphik mit mindestens einer Funktion verknüpft wird.

6. Verfahren nach Anspruch 5, wobei die parametrisierte logische Komponente einen Parameter für ei-

ne Formatzuweisung, Farbzuzuweisung und/oder einer Regelverknüpfung aufweist.

7. Verfahren nach Anspruch 6, wobei die Regelverknüpfung eine funktionale Verknüpfung mit einer Datenbank ist.

8. Verfahren nach einem der Ansprüche 1 bis 7, wobei die Basisvorlage eine von einem Standard-Publishing-Programm erstellte Druckvorlage ist.

9. Verfahren zum Variieren oder Vervollständigen einer nach einem der Ansprüche 1 bis 8 erstellten Druckvorlage durch

Bereitstellen der Druckvorlage auf einem Applikationsserver (54) in einem Datennetz, vorzugsweise dem Internet,

Einwählen (21) eines Benutzers in den Applikationsserver, insbesondere unter Überprüfung (22) einer Zugangsberechtigung und

Zuordnen (23, 24) von Substitutionselementen zu parametrisierten Komponenten durch einen Benutzer oder Client.

10. Vorrichtung zum Erstellen von variablen Druckvorlagen mit

einer Analyseeinrichtung zum Erkennen mindestens eines graphischen Strukturmerkmals (31 bis 37) einer logisch unstrukturierten und graphisch strukturierten Basisdruckvorlage, welche in Form eines Programmcodes vorliegt,

einer Syntheseeinrichtung zum Erzeugen mindestens einer logischen Komponente auf der Grundlage des mindestens einen graphischen Strukturmerkmals und

einer Editoreinrichtung zum Erstellen einer variablen Druckvorlage unter Verwendung der mindestens einen logischen Komponente.

11. Vorrichtung nach Anspruch 10, wobei das mindestens eine graphische Strukturmerkmal die graphische Darstellung einer logischen Komponente wie die eines Wortes, einer Überschrift, einer Textspalte, eines Bildes oder dergleichen ist.

12. Vorrichtung nach Anspruch 11, wobei die Komponente durch ein Substitutionselement, das individuell erzeugbar oder aus einer Vielzahl auswählbar ist, ersetzbar ist.

13. Vorrichtung nach Anspruch 12, wobei das Substitutionselement selbst eine parametrisierte oder nicht

parametrisierte Druckvorlage ist.

14. Vorrichtung nach einem der Ansprüche 10 bis 13, wobei die mindestens eine logische Komponente in der Syntheseeinrichtung parametrisierbar ist, indem die dem mindestens einen Strukturmerkmal zugrunde liegende Graphik mit mindestens einer Funktion verknüpft wird. 5
15. Vorrichtung nach Anspruch 14, wobei die parametrisierte logische Komponente einen Parameter für eine Formatzuweisung, Farbuweisung und/oder einer Regelverknüpfung aufweist. 10
16. Vorrichtung nach Anspruch 15, wobei die Regelverknüpfung eine funktionale Verknüpfung mit einer Datenbank (55) ist. 15
17. Vorrichtung nach einem der Ansprüche 10 bis 16, wobei die Basisvorlage eine von einer Standard-Publishing-Einrichtung erstellte Druckvorlage ist. 20
18. System zum Variieren oder Vervollständigen einer mit einer Vorrichtung nach einem der Ansprüche 10 bis 17 erstellten Druckvorlage mit 25
 - einer Applikationsservereinrichtung (54) zum Bereitstellen der Druckvorlage in einem Daten-
netz (53), vorzugsweise dem Internet,
 - einer Clienteinrichtung (51) zum Einwählen eines Benutzers in den Applikationsserver (54), insbesondere unter Überprüfung einer Zu-
gangsberechtigung, wobei 30
 - Substitutionselemente zu parametrisierten logischen Komponenten durch einen Benutzer oder Client von der Clienteinrichtung (51) aus zuordenbar sind. 35
19. Computerprogramm, das ein Verfahren nach einem der Ansprüche 1 bis 9 implementiert. 40
20. Datenträger, auf dem ein Computerprogramm nach Anspruch 19 gespeichert ist. 45
21. Computersystem, in dem ein Computerprogramm nach Anspruch 19 gespeichert und lauffähig ist. 50

Claims

1. Method for producing variable printing patterns by providing a logically unstructured and graphically structured basic printing pattern which is present in the form of a program code, 55
 - automatically detecting (12) at least one graph-

ical structural feature (31 to 37) of the basic printing pattern, automatically producing (13) at least one logical component based on the at least one graphical structural feature (31 to 37), and automatically producing (15) a variable printing pattern by using the at least one logical component.

2. Method according to claim 1, wherein the at least one graphical structural feature (31 to 37) is the graphical representation of a logical component as that of a word, a title, a text column, a picture or the like.
3. Method according to claim 2, wherein the logical component is replaced by a substitution element, which is generated individually or is selectable from a plurality.
4. Method according to claim 3, wherein the substitution element itself is a parameterized or a non-parameterized printing pattern.
5. Method according to any of the claims 1 to 4, wherein the at least one logical component is parameterized by linking the graphic underlying the at least one structural feature to at least one function.
6. Method according to claim 5, wherein the parameterized logical component has a parameter for a format allocation, color allocation and/or rule linkage.
7. Method according to claim 6, wherein the rule linkage is a functional linkage to a database.
8. Method according to any of the claims 1 to 7, wherein the basic pattern is a printing pattern produced by a standard publishing program.
9. Method for varying or completing a printing pattern produced according to any of the claims 1 to 8 by providing the printing pattern on an application server (54) in a data network, preferably the Internet,
 - login (21) of a user into the application server, especially with verification (22) of an access authorization, and
 - associating (23, 24) substitution elements with parameterized components by a user or client.
10. Device for producing variable printing patterns comprising an analyzing means for detecting at least one graphical structural feature (31 to 37) of a logically unstructured and graphically structured basic printing pattern which is present in the form of a program code,

a synthesizing means for producing at least one logical component based on the at least one graphical structural feature, and
an editor means for producing a variable printing pattern by using the at least one logical component.

11. Device according to claim 10, wherein the at least one graphical structural feature is the graphical representation of a logical component as that of a word, a title, a text column, a picture or the like. 10
12. Device according to claim 11, wherein the component is replaceable by a substitution element, which is generable individually or is selectable from a plurality. 15
13. Device according to claim 12, wherein the substitution element itself is a parameterized or a non-parameterized printing pattern. 20
14. Device according to any of the claims 10 to 13, wherein the at least one logical component is parameterizable in the synthesizing means by linking the graphic underlying the at least one structural feature to at least one function. 25
15. Device according to claim 14, wherein the parameterized logical component has a parameter for a format allocation, color allocation and/or rule linkage. 30
16. Device according to claim 15, wherein the rule linkage is a functional linkage to a database (55).
17. Device according to any of the claims 10 to 16, wherein the basic pattern is a printing pattern produced by a standard publishing means. 35
18. System for varying or completing a printing pattern produced with a device according to any of the claims 10 to 17, comprising 40
 - an application server means (54) for providing the printing pattern in a data network (53), preferably the Internet, 45
 - a client means (51) for the login of a user into the application server (54), especially with verification of an access authorization, wherein substitution elements can be associated with parameterized logical components by a user or client from the client means (51). 50
19. Computer program, which implements a method according to any of the claims 1 to 9.
20. Data carrier, on which a computer program according to claim 19 is stored.

21. Computer system, in which a computer program according to claim 19 is stored and able to run.

5 Revendications

1. Procédé de réalisation de maquettes d'impression variables par:
 - mise à disposition d'une maquette de base non structurée logiquement et structurée graphiquement qui se présente sous forme d'un code de programme,
 - détection automatique (12) d'au moins une caractéristique de structure graphique (31 à 37) de la maquette de base,
 - génération automatique (13) d'au moins une composante logique sur la base d'au moins une caractéristique de structure (31 à 37) et
 - génération automatique (15) d'une maquette variable de la au moins une composante logique.
2. Procédé selon la revendication 1, dans lequel la au moins une caractéristique de structure graphique (31 à 37) est la représentation graphique d'une composante logique telle que celle d'un mot, d'un titre, d'une colonne de texte, d'une image, etc.
3. Procédé selon la revendication 2, dans lequel la composante logique est remplacée par un élément de substitution pouvant être généré de manière individuelle ou pouvant être sélectionné à partir d'une multitude.
4. Procédé selon la revendication 3, dans lequel l'élément de substitution lui-même est une maquette paramétrée ou non paramétrée.
5. Procédé selon l'une des revendications 1 à 4, dans lequel la au moins une composante logique est paramétrée en combinant la graphique à l'origine de la au moins une caractéristique de structure avec au moins une fonctionnalité.
6. Procédé selon la revendication 5, dans lequel la composante logique paramétrée présente un paramètre d'affectation de format, de couleur et/ou du couplage logique de règles.
7. Procédé selon la revendication 6, dans lequel le couplage logique de règles est une liaison fonctionnelle avec une base de données.
8. Procédé selon l'une des revendications 1 à 7, dans lequel la maquette de base est une maquette établie par le programme de publication standard.

9. Procédé destiné à varier ou à compléter une maquette réalisée selon une des revendications 1 à 8 par

mise à disposition de la maquette sur un serveur d'application (54) à l'intérieur d'un réseau informatique, de préférence sur l'Internet, entrée (21) d'un utilisateur dans le serveur d'application, notamment en vérifiant (22) son autorisation d'accès et affectation (23,24) d'éléments de substitution à des composantes paramétrées effectuée par un utilisateur ou client.

10. Dispositif de réalisation de maquettes variables comprenant :

un dispositif d'analyse destiné à la détection d'au moins une caractéristique de structure graphique (31 à 37) d'une maquette de base non structurée logiquement et structurée graphiquement qui se présente sous forme d'un code de programme ;
un dispositif de synthèse destiné à générer au moins une composante logique sur la base de la au moins une caractéristique de structure graphique et
un dispositif d'édition destiné à la réalisation d'une maquette variable en utilisant la au moins une composante logique.

11. Dispositif selon la revendication 10, dans lequel la au moins une caractéristique de structure graphique est la représentation graphique d'une composante logique telle que celle d'un mot, d'un titre, d'une colonne de texte, d'une image, etc.

12. Dispositif selon la revendication 11, dans lequel la composante peut être remplacée par un élément de substitution susceptible d'être généré individuellement ou d'être sélectionné à partir d'une multitude.

13. Dispositif selon la revendication 12, dans lequel l'élément de substitution lui-même est une maquette paramétrée ou non paramétrée.

14. Dispositif selon l'une des revendications 10 à 13, dans lequel la au moins une composante logique peut être paramétrée par le dispositif de synthèse en combinant la graphique à l'origine de la au moins une caractéristique de structure avec au moins une fonctionnalité.

15. Dispositif selon la revendication 14, dans lequel la composante logique paramétrée présente un paramètre d'affectation du format, de la couleur et/ou du couplage logique de règles.

16. Dispositif selon la revendication 15, dans lequel le couplage logique de règles est une liaison fonctionnelle avec une base de données (55).

17. Dispositif selon l'une des revendications 10 à 16, dans lequel la maquette de base est une maquette réalisée par un dispositif de publication standard.

18. Système destiné à varier ou à compléter une maquette réalisée par un dispositif selon une des revendications 10 à 17 et comprenant :

un serveur d'application (54) prévu pour la mise à disposition de la maquette sur un réseau informatique (53), notamment l'Internet, un dispositif de client (51) destiné à entrer un utilisateur dans le serveur d'application (54), notamment en vérifiant son autorisation d'accès, dans lequel des éléments de substitution à des composantes logiques paramétrées peuvent être affectés par un utilisateur ou un client à partir du dispositif de client (51).

19. Programme informatique pour PC mettant en oeuvre le procédé selon une des revendications 1 à 9.

20. Support de données destiné au stockage du programme informatique selon la revendication 19.

21. Système d'ordinateur dans lequel est stocké un programme informatique selon la revendication 19 et dans lequel il est apte à fonctionner.

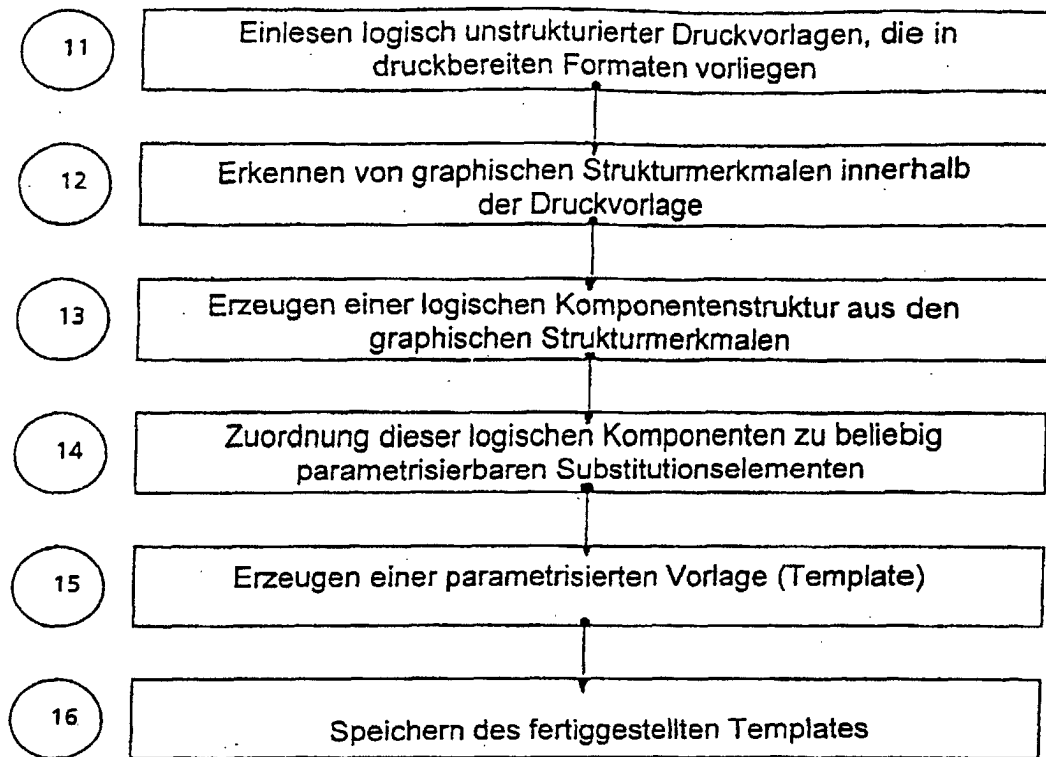


Fig.1

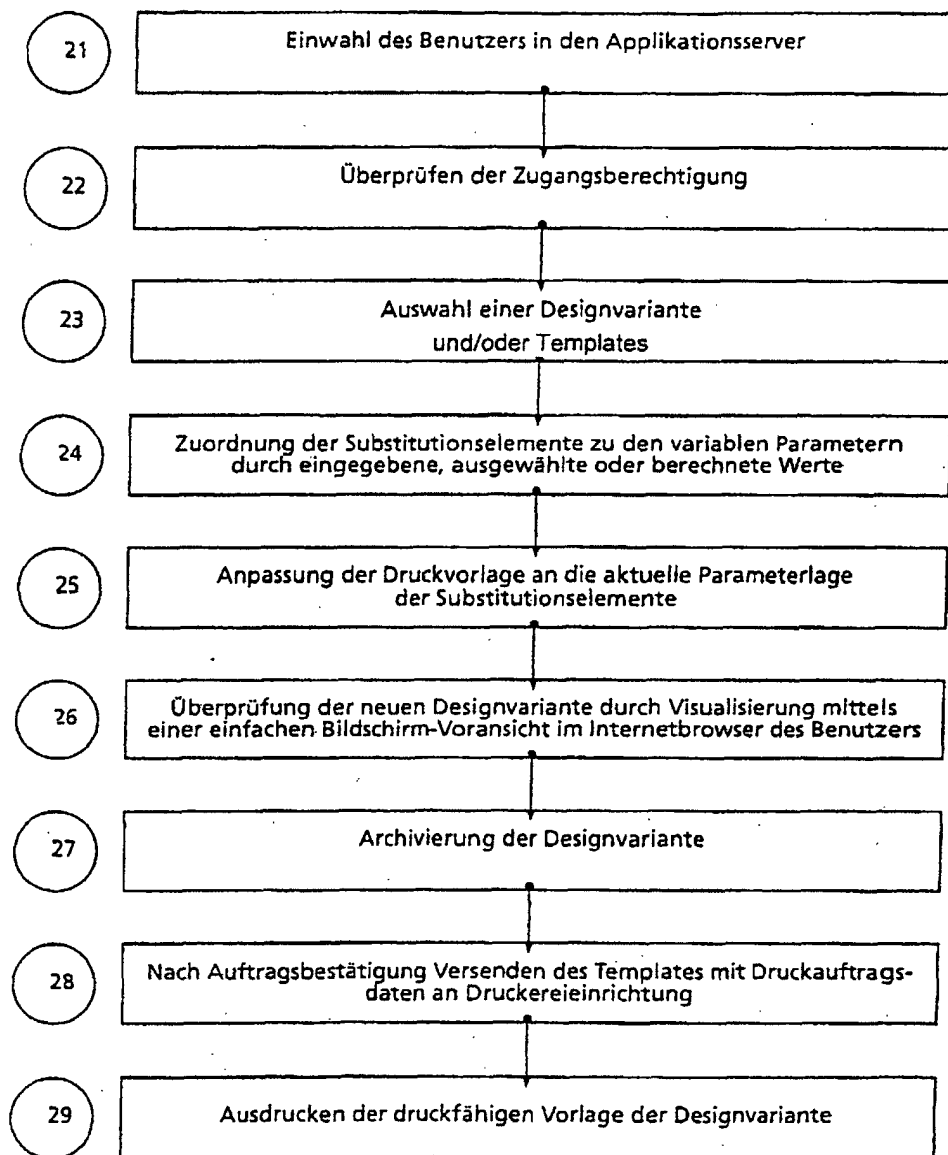


Fig.2

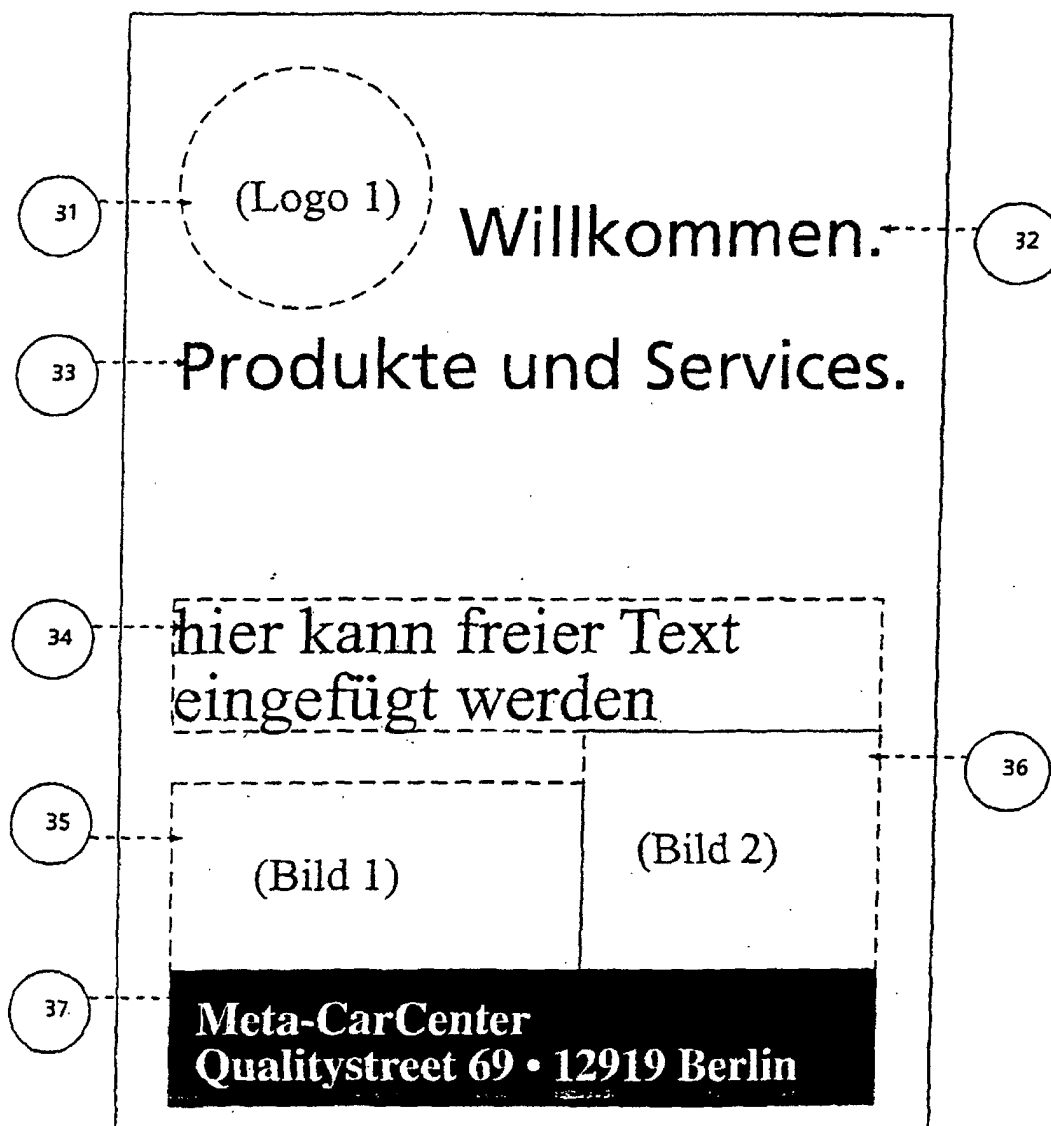


Fig.3

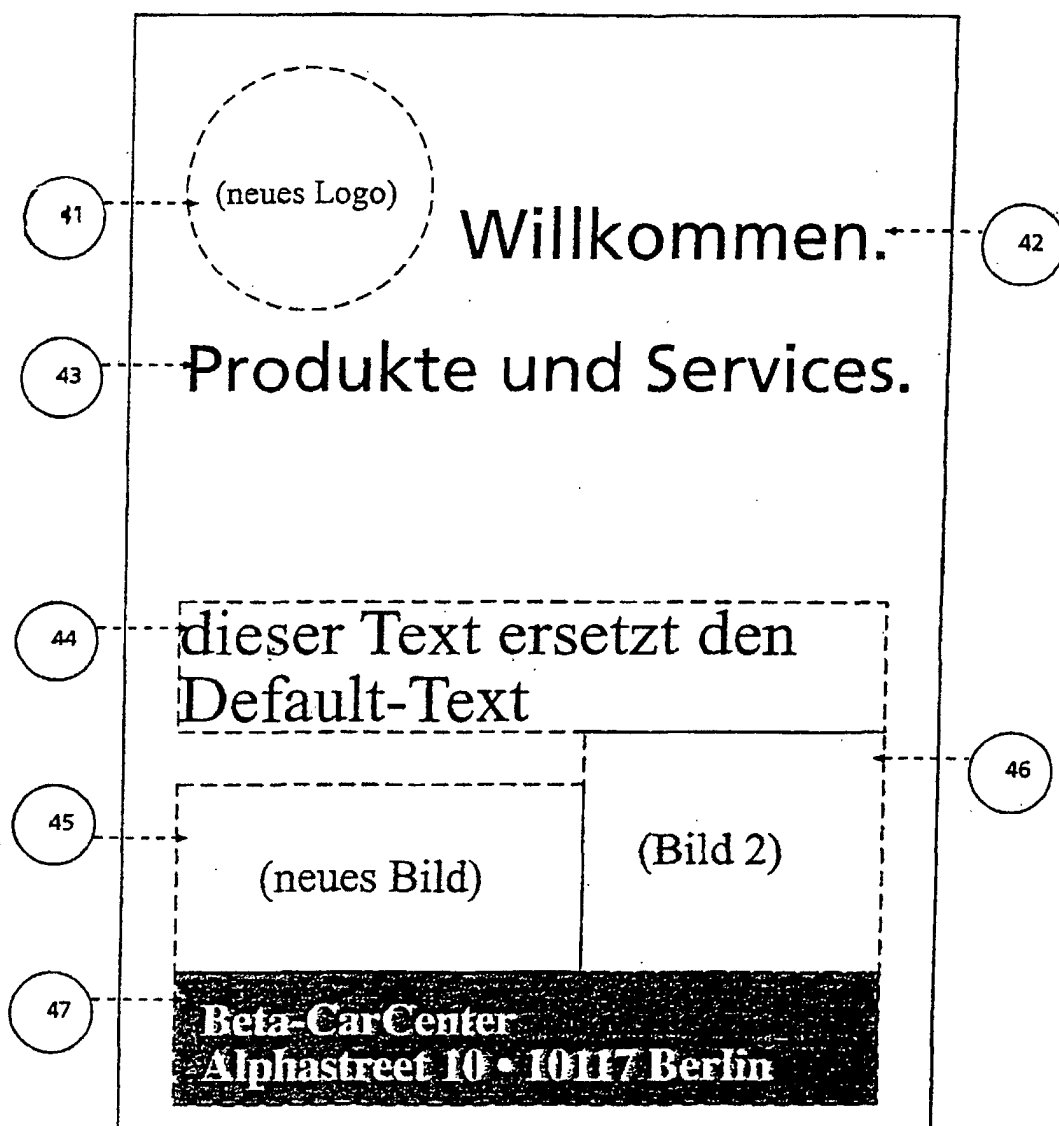


Fig.4

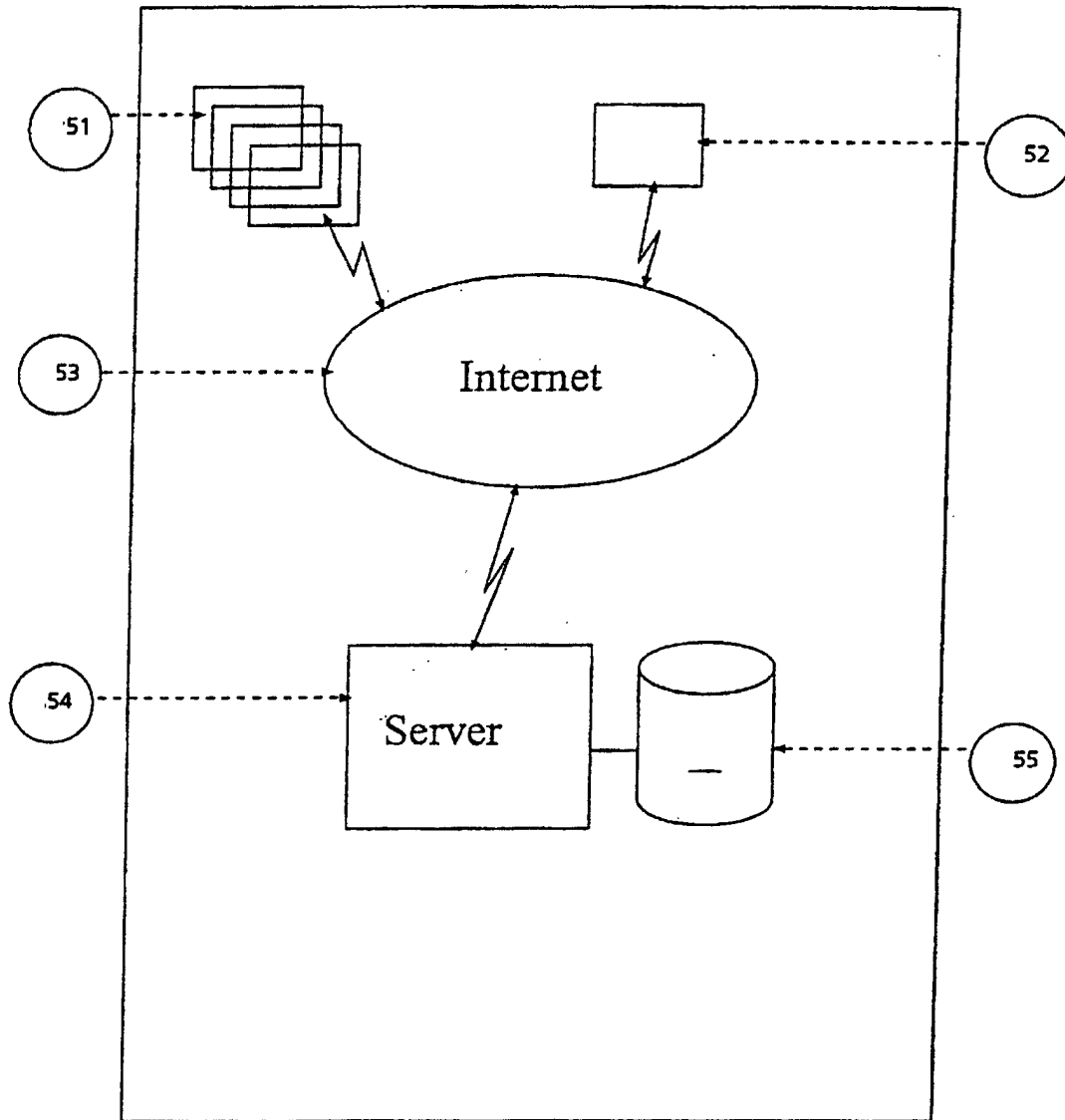


Fig.5