



(10) **DE 10 2011 014 625 B4** 2015.11.12

(12)

## Patentschrift

(21) Aktenzeichen: **10 2011 014 625.3**  
(22) Anmeldetag: **21.03.2011**  
(43) Offenlegungstag: **27.09.2012**  
(45) Veröffentlichungstag  
der Patenterteilung: **12.11.2015**

(51) Int Cl.: **H04N 21/20** (2011.01)  
**H04N 21/234** (2011.01)  
**H04N 21/2668** (2011.01)  
**H04N 21/47** (2011.01)

Innerhalb von neun Monaten nach Veröffentlichung der Patenterteilung kann nach § 59 Patentgesetz gegen das Patent Einspruch erhoben werden. Der Einspruch ist schriftlich zu erklären und zu begründen. Innerhalb der Einspruchsfrist ist eine Einspruchsgebühr in Höhe von 200 Euro zu entrichten (§ 6 Patentkostengesetz in Verbindung mit der Anlage zu § 2 Abs. 1 Patentkostengesetz).

(73) Patentinhaber:  
**Mackevision Medien Design GmbH Stuttgart,  
70174 Stuttgart, DE**

(74) Vertreter:  
**DREISS Patentanwälte PartG mbB, 70174  
Stuttgart, DE**

(72) Erfinder:  
**Boy, Simon, 70188 Stuttgart, DE**

(56) Ermittelter Stand der Technik:

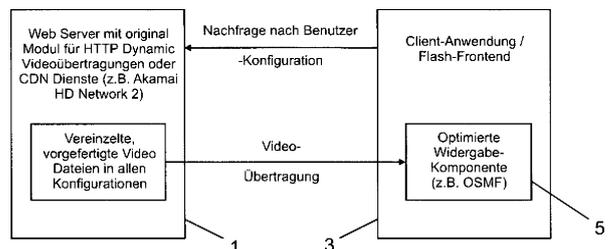
**DE 10 2005 059 992 A1**  
**US 2007 / 0 005 795 A1**  
**US 2008 / 0 071 645 A1**  
**US 4 305 131 A**  
**WO 2010/ 019 925 A1**

(54) Bezeichnung: **Verfahren zum Bereitstellen eines Videofilms mit mindestens einem während des Ablaufs konfigurierbaren Objekts**

(57) Hauptanspruch: Verfahren zum Bereitstellen eines Video-Films, bei dem mindestens ein dargestelltes Objekt während des Ablaufs konfigurierbar ist, umfassend die Verfahrensschritte:

Erstellen einer Vielzahl von Video-Filmen, die verschiedene Konfigurationen des konfigurierbaren Objekts enthalten, Speichern der Video-Filme unter einem Dateinamen (LIM ROT 01, COU ROT 01, LIM WEI 01, COU WEI 01,) so dass jeder Video-Film in mehrere Fragmente (9.1.1 bis 9.10.4) unterteilt ist und

Bereitstellen eines individuellen Video-Films aus einer Folge von aneinandergehängten Fragmenten (9.1.2, 9.2.2, 9.3.2, 9.4.4, 9.5.4, 9.6.4, 9.7.4, 9.8.4, 9.9.3, 9.10.3), wobei immer dann von einem momentan ablaufenden Video-Film mit einer vom Nutzer eines Client-PC (3) bestimmten Konfiguration des konfigurierbaren Objekts zu einem anderen Video-Film mit einer anderen Konfiguration des konfigurierbaren Objekts gewechselt wird, wenn die Konfiguration des dargestellten Objekts vom Nutzer des Client-PC (3) geändert wird.



## Beschreibung

**[0001]** In den Internetpräsenzen von Herstellern hochwertiger Güter wird dem potenziellen Kunden, der über seinen Client-PC auf die Internetpräsenz des Herstellers zugreift, häufig die Möglichkeit geboten, bestimmte Produkte seinen Wünschen entsprechend zu konfigurieren. Ein typisches Beispiel für ein solches vom User des Client-PC konfigurierbares Produkt ist ein PKW. Bei PKWs gibt es verschiedene Ausstattungsmerkmale, wie z. B. die Farbe der Karosserie, die Art und Form der Räder sowie Karosserievarianten (Limousine, Kombi oder Coupé). Diese Varianten können miteinander kombiniert werden, sodass sich eine Vielzahl von möglichen Konfigurationen des Produkts ergibt. Solche Konfiguratoren sind seit einigen Jahren bereits üblich und erlauben es dem potentiellen Kunden einen PKW in der von ihm gewünschten Konfiguration am PC anzusehen. Ein nächster Schritt besteht nun darin, dem potenziellen Kunden einen visuellen Eindruck des von dem Kunden konfigurierten Produkts in einem Videofilm unter möglichst realistischen Bedingungen zu vermitteln. Dies kann z. B. dadurch erfolgen, dass ein PKW in der vom Kunden erstellten Konfiguration in einem Videofilm auf einer Landstraße fahrend dargestellt wird und der potentielle Kunde sich diesen Videofilm an seinem PC anschauen kann. Dadurch werden z. B. die Farbe der Karosserie im Zusammenspiel mit der Umgebung und den Lichtreflexen sehr anschaulich.

**[0002]** Wenn nun der potenzielle Kunde, der nachfolgend auch als User des Client-PCs bezeichnet wird, während dem Ablauf dieses Videofilms die Konfiguration des PKWs ändern will, so ist dies bislang nicht möglich.

**[0003]** Aus der US 2007/0005795 A1 ist ein Verfahren bekannt, bei dem während der Laufzeit eines „Data Streams“ der User Objekte dieses „Data Streams“ anzeigen, ausblenden oder sogar animieren kann.

**[0004]** Aus der US 4,305,131 ist ein Verfahren zum Abspielen eines Videofilms bekannt, bei dem der Betrachter an vorbestimmten Stellen des Videofilms durch Sprachbefehle den weiteren Ablauf der in dem Videofilm dargestellten Geschichte steuern kann.

**[0005]** Der Erfindung liegt die Aufgabe zu Grunde, ein Verfahren bereit zu stellen, mit dessen Hilfe es möglich ist, dem Nutzer eines Client-PCs einen Videofilm zur Verfügung zu stellen, bei dem während des Ablaufs die Konfiguration mindestens eines dargestellten Objekts, z. B. eines PKWs, zu jedem Zeitpunkt und beliebig oft geändert werden kann.

**[0006]** Diese Aufgabe wird erfindungsgemäß gelöst durch ein Verfahren nach dem Anspruch 1.

**[0007]** Immer dann, wenn ein neues Fragment des momentan ablaufenden Videofilms in das Videoprojekt des Client-PCs geladen wird, sieht das erfindungsgemäße Verfahren die Möglichkeit vor, dieses neu zu ladende Fragment von einem anderen der abgespeicherten Videofilme zu laden und dadurch die in diesem anderen Videofilm dargestellte Konfiguration des Objekts zu übernehmen.

**[0008]** Dadurch ist es möglich, z. B. in den ersten 5 Sekunden eines Videofilms, entsprechend den ersten 5 Fragmenten eines Videofilms, ein Fahrzeug mit weißer Karosserie in dem Videofilm darzustellen. Wenn nun der User des Client-PCs auf dem der Videofilm abläuft, beispielsweise nach 5,2 Sekunden entscheidet, dass er den Film nun mehr mit einem PKW mit roter Karosserie weiter betrachten will, dann wird beim Laden des sechsten Fragments vom einem Server auf den Client-PC nach etwa 6 Sekunden, dieses sechste Fragment nicht von dem Videofilm in dem ein PKW mit weißer Karosserie dargestellt ist geladen, sondern von dem Videofilm in dem die Karosseriefarbe Rot ist.

**[0009]** Dadurch, dass auf dem Server oder einem Server-Netzwerk von allen verfügbaren Konfigurationen, wie z. B. Karosserievarianten, Farben und Rädervarianten, jeweils ein Videofilm abgespeichert ist, ist es möglich, beim Laden des jeweils nächsten Fragments auf einen anderen Videofilm zu wechseln, der eine andere Konfiguration des dargestellten Objekts darstellt. Dabei läuft der Videofilm für den User des Client-PCs ohne Verzögerung und in unveränderter Qualität ab.

**[0010]** Das erfindungsgemäße Verfahren ist verhältnismäßig einfach zu realisieren und ermöglicht dennoch eine Vielzahl von verschiedenen Konfigurationen eines dargestellten Objekts beim einmaligen Ablauf eines Videofilms darzustellen. Dabei wird die Internetverbindung zwischen Server und Client-PC nicht mehr belastet, als wenn ein „durchgängiger“ Videofilm dargestellt würde. Dies liegt darin begründet, dass bei dem „normalen“ Abspielen eines Videofilms nicht der gesamte Film auf einmal vom Server auf den Client-PC übertragen wird, sondern in Form von Fragmenten erfolgt.

**[0011]** Der Wechsel von einem Videofilm zum Anderen kann vorteilhafterweise mit Hilfe der Adobe HTTP Dynamic-Streaming-Software erfolgen. Diese Dynamic-Streaming-Software umfasst eine „adaptive Bitrate“ genannte Funktion. Diese Funktion wird dazu benutzt, um bei schwankender Leistungsfähigkeit der Internetverbindung zwischen Client-PC und Server die Auflösung, mit der der Videofilm auf dem client-PC dargestellt wird, der aktuellen Leistungsfähigkeit der Datenübertragung anzupassen. So kann von einem Film mit einer hohen Auflösung und entsprechend großen Bitrate, auf einen Film mit gerin-

gerer Auflösung und entsprechend kleinerer Bitrate gewechselt werden, wenn die Leitungsfähigkeit der Internetverbindung dies erforderlich macht. Generell wird diese Technologie "Adaptives HTTP-Streaming" genannt. Diese Technologie wird auch von anderen Herstellern für weitere Zielplattformen angeboten. Hier wäre das "Apple HTTP Live Streaming" für QuickTime, iPhone oder iPad und "Microsoft IIS Smooth Streaming" für zum Beispiel den Silverlight Player oder Xbox 360 zu erwähnen. Alle basieren technisch auf dem selben Prinzip wie Adobes HTTP Dynamic Streaming. Daher ist das erfindungsgemäße Verfahren darauf auch anwendbar. In einer besonders vorteilhaften Ausgestaltung des erfindungsgemäßen Verfahrens ist vorgesehen, diese Funktion dazu zu verwenden, um zwischen verschiedenen Filmen, die verschiedene Konfigurationen eines dargestellten Objekts, wie z. B. eines PKWs, enthalten, hin und her zu springen und zwar während der Film abläuft. Dabei bleibt die Qualität der Darstellung konstant.

**[0012]** Um ein möglichst rasches Wechseln der in dem ablaufenden Videofilm dargestellten Konfiguration des konfigurierbaren Objekts zu ermöglichen, ist vorgesehen, dass die Fragmente der Videofilme Zeitabschnitte mit einer Dauer von weniger als 4 Sekunden, bevorzugt von weniger als 2 Sekunden und besonders bevorzugt gleich einer Sekunde, umfassen. Mit dieser kurzen Fragmentierung ist ein rascher Wechsel von einer Konfiguration zur nächsten Konfiguration möglich. In der Praxis kann dadurch der Wechsel innerhalb von weniger als 2 Sekunden erreicht werden.

**[0013]** Zusätzlich oder alternativ zu der Fragmentierung der Videofilme nach einem starren Zeitraster (von z. B. 1 s Dauer) ist es auch möglich, dass ein Fragment der Videofilme mit dem Beginn einer Szene des Videofilms anfängt und mit dem Ende einer Szene des Videofilms endet. Dann nämlich muss ohnehin ein komplett neuer Bildaufbau vorgenommen werden, sodass der Wechsel von einer Konfiguration zu einer anderen Konfiguration besonders einfach möglich ist und die Datenübertragung zwischen Server und Client-PC optimiert wird.

**[0014]** Um einen einfachen Wechsel der Konfiguration durch den User des Client-PCs zu ermöglichen, ist vorgesehen, dass die Konfiguration des dargestellten Objekts in einem Konfigurationscode enthalten ist und dieser Konfigurationscode vom Client-PC an den Server übertragen wird. Wenn sich dieser Konfigurationscode während des Ablaufs des Videofilms ändert, dann wird auch diese Information vom Client-PC auf den Server übermittelt und entsprechend dem Konfigurationscode greift der Server dann auf einen anderen Videofilm zu, wenn er das nächste Fragment des Videofilms an den Client-PC übermittelt.

**[0015]** Um einen übersichtlichen Konfigurationscode zu erhalten, ist vorgesehen, dass dieser verschiedene Abschnitte umfasst, wobei einige dieser Abschnitte zur Beschreibung der Form, der Farbe und/oder der Ausstattung des konfigurierbaren Objekts dienen. Damit ist es beispielsweise möglich eine Karosserieform, die Farbe der Karosserie und/oder die gewählten Räder eines PKWs eindeutig zu identifizieren und mit Hilfe dieses Konfigurationscodes auf den entsprechenden Videofilm zuzugreifen zu können.

**[0016]** Es ist selbstverständlich auch möglich, die eigentliche Funktionalität des Adobe HTTP Dynamic Streaming, nämlich die Anpassung der Bitrate auch in das erfindungsgemäße Verfahren zu integrieren. Dann sind zusätzlich noch Filme mit identischem Inhalt; d. h. gleicher Konfiguration des konfigurierbaren Objekts; in verschiedenen Auflösungen und entsprechend verschiedenen Bitraten auf dem Server abgespeichert. Je nachdem welche Bitrate in dem Konfigurationscode angefordert wird, wird dann auf die entsprechende Kopie des Videofilms zugegriffen.

**[0017]** In aller Regel werden alle Videofilme zwischen denen während des Ablaufs gewechselt werden kann, die gleiche Spieldauer haben. Es ist jedoch auch denkbar, dass die Filme verschiedene Längen haben. Beispielsweise wäre es denkbar, dass, wenn ein Cabriolet gezeigt wird, eine Sequenz eingeschoben wird oder am Ende des Videofilms angehängt wird, in dem das Öffnen und Schließen des Verdecks gezeigt wird. Eine solche Sequenz ist selbstverständlich nicht notwendig bei Varianten mit festem Dach.

**[0018]** Die eingangsgenannte Aufgabe wird ebenfalls gelöst durch ein Computerprogramm für einen Computer, das nach einem der erfindungsgemäßen Verfahren abläuft sowie einem Computer der als Internetserver eingesetzt wird.

**[0019]** Weitere Vorteile und vorteilhafte Ausgestaltungen der Erfindung sind der nachfolgenden Zeichnung, deren Beschreibung und den Patentansprüchen entnehmbar. Alle in der Zeichnung, deren Beschreibung und den Patentansprüchen genannten Merkmale können sowohl einzeln als auch in beliebiger Kombination miteinander erfindungswesentlich sein.

Zeichnung

**[0020]** Es zeigen:

**[0021]** Fig. 1 eine grundsätzliche Anordnung von Server und Client-PC und die zwischen beiden ausgetauschten Daten,

**[0022]** Fig. 2 ein Beispiel eines Konfigurationscodes und

**[0023]** Fig. 3 und Fig. 4 den erfindungsgemäßen Ablauf eines Videofilms, bei dem während des Ablaufs die Konfiguration mindestens eines in dem Videofilm dargestellten Objekts geändert wird.

#### Beschreibung der Ausführungsbeispiele

**[0024]** In der Fig. 1 ist stark vereinfacht die Kommunikation zwischen einem Server **1** und einem Client-PC **3** dargestellt. Der Server **1** kann der Webserver des Unternehmens sein, dessen Produkt in dem Videofilm dargestellt werden soll. Er ist bevorzugt mit dem Adobe HTTP Dynamic Streaming ausgestattet. Es kann aber auch eine andere Streaming-Technologie verwendet werden, auf welche die Beschreibung "Adaptives HTTP-Streaming" zutrifft, um zum Beispiel auch andere Zielplattformen wie Apple iPhone/iPad zu bedienen. Das HTTP Dynamic Streaming ist eine von der Firma Adobe bereitgestellte Software, die auch die Anpassung der Bitrate an die Leistungsfähigkeit der Internetverbindung zwischen Server **1** und Client-PC **3** ermöglicht. Diese Funktionalität wird als „adaptive Bitrate“ bezeichnet.

**[0025]** Alternativ zu dem Webserver eines Unternehmens kann der Server **1** auch ein „content delivery network service“ (nachfolgend CDN-Service) sein. Ein Beispiel für einen solchen CDN-Service ist das Akamai HD Network **2**.

**[0026]** Auf dem Server **1** ist eine Vielzahl vorgefertigter und fragmentierter Videodateien abgespeichert, wobei zu jeder möglichen Konfiguration des konfigurierbaren Objekts eine Videodatei abgespeichert ist.

**[0027]** Dies bedeutet, dass, nachdem der User eines Client-PCs **3** eine Konfiguration des in dem Videofilm dargestellten Objekts ausgewählt hat, die diese Konfiguration enthaltende Videodatei ausgesucht wird, und diese Videodatei vom Server **1** an den Client-PC **3** übermittelt wird.

**[0028]** Wenn ein solcher Film nicht während des Ablaufs konfigurierbar ist, dann wird einmal vor dem Start des Videofilms vom Client-PC **3** die Information über die vom User ausgewählte Konfiguration (nachfolgend als Konfigurationscode bezeichnet) an den Server übermittelt. In Abhängigkeit des Konfigurationscodes wählt der Server die zugehörige Videodatei aus und übermittelt den Videofilm an den Client-PC. Dies geschieht dadurch, dass ein Fragment des Videofilms nach dem anderen vom Server **1** an den Client-PC **3** mit Hilfe von zum Beispiel einer Internetverbindung übermittelt wird. Der Client-PC hat ein sogenanntes Videobjekt **5**, welches das Abspielen des vom Server **1** übermittelten Videofilms (HTTP-Video-Stream) ermöglicht.

**[0029]** Das Videobjekt **5** des Client-PCs **3** kann ein sogenanntes Open Source Media Framework

(nachfolgend OSMF) sein, wie es beispielsweise von der Firma Adobe bereitgestellt wird. Das Videobjekt **5** umfasst einen Video-Puffer (ohne Bezugszeichen) in dem die vom Server **1** geschickten Fragmente nacheinander zwischengespeichert werden. Aus dem Fragment, welches sich im Videopuffer des Videobjekts befindet, werden die auf dem Bildschirm dargestellten Bildsequenzen des Videofilms erzeugt.

**[0030]** Um ein möglichst rasches Umschalten von einem Videofilm mit einer ersten Konfiguration des dargestellten Objekts zu einem Videofilm mit einer zweiten Konfiguration des dargestellten Objekts zu ermöglichen, ist in erfindungsgemäßer Weise vorgesehen, den Videopuffer des Videobjekts so einzustellen, dass er nur eine kurze Sequenz, beispielsweise von einer Sekunde Dauer abspeichert. Die Pufferkapazität des Videopuffers ist idealerweise so auf die Länge der vom Server **3** übermittelten Fragmente abgestimmt, dass immer jeweils ein Fragment in dem Videopuffer zwischengespeichert wird.

**[0031]** Dadurch, dass erfindungsgemäß vorgesehen ist, Fragmente von nur einer Sekunde Dauer vom Server **1** an den Client-PC **3** zu übermitteln und in dem Videopuffer des Client-PCs **3** nur Fragmente von ebenfalls einer Sekunde Dauer zwischenzuspeichern ist es möglich, dass auf dem Bildschirm des Client-PCs **3** nach maximal zwei Sekunden Dauer das Objekt in einer anderen Konfiguration erscheint, wenn der User des Client-PCs **3** dies wünscht. Dies bedeutet, dass die Befehle des Nutzers des Client-PCs **3** sehr rasch umgesetzt werden.

**[0032]** Die Kommunikation zwischen Client-PC **3** und Server **1** erfolgt über handelsübliche oder allgemein zugängliche Softwarepakete. Daher wird diese Kommunikation nicht im Detail erläutert. Wichtig ist jedoch zu erwähnen, dass der Client-PC **3** eine Client Application beziehungsweise ein Flash Front End aufweist, die es dem Nutzer ermöglicht, aus der Vielzahl möglicher Konfigurationen des dargestellten Objekts eine Auswahl zu treffen. Anhand dieser Auswahl wird ein entsprechender Konfigurationscode vom Client-PC **3** generiert und über die Internetverbindung an den Server **3** übermittelt. Das erfindungsgemäße Verfahren kann mit dem sogenannten Adobe Flash Softwarepaket ausgeführt werden, wobei dieses in gewissem Maße geändert/adaptiert werden muss. Das Adobe Flash Softwarepaket ist eine weit verbreitete Technologie, um interaktive Webseiten zu generieren.

**[0033]** Des Weiteren wird bei der Datenübertragung zwischen dem Server **1** und dem Client-PC **3** die sogenannte Adobe HTTP Dynamic Streaming Technologie mit "adaptiver Bitrate" eingesetzt. Erfindungsgemäß ist nun vorgesehen, die Funktionalität der „adaptiven Bitrate“ zu modifizieren und dazu einzusetzen,

dass zwischen verschiedenen Videofilmen mit unterschiedlichen Inhalten hin und her geschaltet wird.

**[0034]** Wie bereits mehrfach erwähnt, enthält jeder Videofilm einen Film bei dem das konfigurierbare Objekt in einer anderen Konfiguration dargestellt ist. Durch das Hin- und Herspringen von einem Film zum anderen Film kann somit während des Ablaufs des Videofilms die Konfiguration geändert werden und das Ergebnis dieser geänderten Konfiguration wird unmittelbar und ohne Unterbrechungen, ohne Ruckeln oder andere Unannehmlichkeiten am Bildschirm des Client-PC **3** dargestellt.

**[0035]** In **Fig. 2** ist beispielhaft der Aufbau eines Konfigurationscodes dargestellt. Selbstverständlich ist es möglich, einen Konfigurationscode auch ganz anders zu realisieren.

**[0036]** In **Fig. 2** ist ein Beispiel eines neunstelligen Konfigurationscodes „LIMROT01“ zur Erläuterung dargestellt. Dieser achtstellige Konfigurationscode **7** hat einen ersten Abschnitt **I**, einen zweiten Abschnitt **II**, einen dritten Abschnitt **III** und optional noch einen vierten Abschnitt **IV**.

**[0037]** Wenn das konfigurierbare und in einem Videofilm dargestellte Objekt beispielsweise ein PKW ist, dann könnte der erste Abschnitt **I** des Konfigurationscodes beispielsweise eine Karosserievariante beschreiben. Wenn es von einem PKW eine Limousine, ein Kombi und ein Coupé gibt, gäbe es demzufolge im ersten Abschnitt **I** insgesamt drei verschiedene Codes, beispielsweise LIM, COU und KOM.

**[0038]** Der zweite Abschnitt **II** könnte zur Definition der Farbe der Karosserie dienen. In diesem Fall würde eine in dem zweiten Abschnitt **II** über die Ziffern- oder Zahlenfolge des zweiten Abschnitts **II** eine Zahl aus einer zuvor definierten Liste ausgewählt. So könnte eine Vielzahl von Farben in dem zweiten Abschnitt **II** definiert/konfiguriert werden. Diese Liste kann beliebig vergrößert werden. Wenn mehr als 999 Farben zur Auswahl stehen, muss der zweite Abschnitt **II** mehr als drei Stellen aufweisen.

**[0039]** In einem dritten Abschnitt **III** können Ausstattungsvarianten oder verschiedene Räder ausgewählt werden. Wenn der dritte Abschnitt **III** zwei Ziffern hat, können insgesamt 99 Rädervarianten damit identifiziert werden. Selbstverständlich können auch Buchstabencodes verwendet werden oder die Zahl der Stellen erhöht werden, wenn mehr als 99 Varianten konfiguriert werden sollen.

**[0040]** Aus dem Konfigurationscode ist es also möglich, die Karosserievariante, die Karosseriefarbe (Abschnitt **II**) und die ausgewählten Räder (Abschnitt **III**) herauszulesen.

**[0041]** Der Client-PC **3** erstellt in dem Flash Frontend der Client Application entsprechend den Anweisungen des Nutzers einen solchen Konfigurationscode und sendet ihn an den Server **1**.

**[0042]** Immer wenn der Nutzer einen neuen Konfigurationscode erstellt hat und dieser an den Server **1** übermittelt wurde, dann muss bei einem ablaufenden Videofilm auf eine andere Videodatei gewechselt werden, um den Film mit dem "neu" konfigurierten PKW weiterlaufen zu lassen.

**[0043]** Es ist selbstverständlich auch möglich, in dem Konfigurationscode mehr als drei Abschnitte **I**, **II** und **III** vorzusehen. Hier sind prinzipiell keine Grenzen gesetzt.

**[0044]** Es wäre zum Beispiel auch denkbar, dass in einem vierten Abschnitt **IV**, der optional ist, noch die Bitrate beziehungsweise die Qualität der Filme adressiert werden kann, so wäre es beispielsweise möglich, die Funktion der „adaptiven Bitrate“ des Adobe HTTP Dynamic Streaming Software auch in das erfindungsgemäße Verfahren zu integrieren und damit auch bei unterschiedlicher Übertragungsrate zwischen Server **1** und Client-PC **3** auf verschiedene Videofiles zurückzugreifen, die eine unterschiedliche Bitrate erfordern.

**[0045]** In der **Fig. 3** werden die Grundzüge des erfindungsgemäßen Verfahrens an einem Beispiel erläutert. Es wird dabei unterstellt, dass der Film neun Sekunden lang ist. Des Weiteren wird unterstellt, dass aus Gründen der Einfachheit nur zwei Karosserievarianten und zwei Farben zur Verfügung stehen. Somit gibt es vier unterschiedliche Konfigurationen und es müssen vier verschiedene Videofilme im Server **1** abgespeichert werden.

**[0046]** In dem in **Fig. 3** dargestellten Beispiel soll es zwei Karosserievarianten (Limousine (abgekürzt LIM) und Coupé (abgekürzt COU)) geben. Diese Karosserievarianten stehen im ersten Abschnitt **I** des Konfigurationscodes in **Fig. 3**.

**[0047]** Des Weiteren soll es zwei Farben zur Auswahl geben, nämlich rot (ROT) und weiß (abgekürzt WEI) diese Farbcodierung steht im zweiten Abschnitt **II** des Konfigurationscodes.

**[0048]** Im dritten Abschnitt **III**, der das Design der Räder festlegt, sind keine Varianten vorgesehen. Daher ist im dritten Abschnitt **III** immer der Code 01 eingetragen.

**[0049]** Es gibt also  $2 \times 2 = 4$  Konfigurationen, die in der linken Hälfte der **Fig. 3** untereinander aufgelistet sind. Es beginnt in der obersten Zeile mit einer roten Limousine, gefolgt von einem roten Coupé, einer weißen Limousine und einem weißen Coupé. Zu jeder

dieser insgesamt vier Konfigurationen ist eine Videodatei hinterlegt in der das entsprechend konfigurierte Auto durch eine Landschaft fährt.

**[0050]** In dem Beispiel gemäß **Fig. 3** soll der Film 9 Sekunden lang sein. Jede der vier Videodateien ist in Fragmente mit einer Dauer von einer Sekunde (**1s**) unterteilt. Diese Fragmente **9.1.1** bis **9.10.4** sind in **Fig. 3** dargestellt. Aus Gründen der Übersichtlichkeit sind nicht alle Fragmente mit Bezugszeichen versehen.

**[0051]** Als Besonderheit soll bei dem dargestellten Beispiel zum Zeitpunkt  $t = 6,5$  Sekunden ein Szenenwechsel in dem Film stattfinden. Ein solcher Szenenwechsel bedeutet, dass ein vollständig neues Bild aufgebaut werden muss. Diese Notwendigkeit kann bei dem erfindungsgemäßen Verfahren dadurch ausgenutzt werden, um dort einen Konfigurationswechsel vorzunehmen. Deswegen kann es in manchen Anwendungen vorteilhaft sein, zum Zeitpunkt eines Szenenwechsels ein Fragment zu beenden beziehungsweise zu beginnen. Dies ist in dem Zeitintervall zwischen 6 und 7 Sekunden durch eine außerhalb des 1-s-Zeitrasters liegende Fragmentierung zum Zeitpunkt  $t = 6,5$  Sekunden angedeutet. Diese Fragmentierung entsprechend den Szenenwechseln des Films ist optional.

**[0052]** Wenn nun ein Nutzer des Client-PCs **3** den Videofilm sehen will mit einem roten Coupe, dann wird zunächst aus der Datei COU ROT 01 das erste Fragment, welches den Zeitraum von 0 bis 1 Sekunde abdeckt, geladen und vom Server **1** an den Client-PC **3** übermittelt. Solange der Nutzer an dieser Konfiguration nichts ändert, sendet der Server **3** ein Fragment nach dem anderen aus der Datei COU ROT 01 an den Client-PC **3**.

**[0053]** Wenn man unterstellt, dass zum Zeitpunkt  $t = 2,5$  Sekunden der Nutzer nun wissen möchte, wie ein weißes Coupé in dem Film aussieht, dann wählt er eine entsprechende Konfiguration über das Flash Frontend aus. Infolgedessen wird vom Client-PC **3** der Konfigurationscode COU WEI 01 an den Server übermittelt. Dies bedeutet, dass das vierte Fragment aus der Datei COU WEI 01 an den Client-PC übermittelt wird.

**[0054]** Wenn nun beispielsweise nach weiteren 4 Sekunden der Nutzer sehen möchte, wie sich eine weiße Limousine in dem Film macht, dann kann er diese Konfiguration während dem der Film abläuft, vornehmen und es wird vom Client-PC **3** eine entsprechende Konfigurationscode, nämlich LIM WEI 01 an den Server übermittelt. Infolgedessen übermittelt der Server nun ab der Sekunde **7** das nächste Fragment **9.9.3** von der Videodatei LIM WEI 01 an den Client-PC.

**[0055]** In der **Fig. 4** ist dieser beispielhaft anhand der **Fig. 3** erläuterte zeitliche Ablauf durch eine Linie **11** angedeutet.

**[0056]** Im unteren Teil der **Fig. 4** ist der beim Client-PC **3** ankommende Videofilm, der aus den hintereinander gehängten Fragmenten verschiedener Dateien besteht dargestellt. Dabei ist jeweils für den entsprechenden Zeitabschnitt die zugehörige Datei aus dem die Fragmente stammen, eingetragen.

**[0057]** Daraus wird deutlich, dass durch das erfindungsgemäße Verfahren dem Client-PC **3** eine Serie von Fragmenten übermittelt wird, die aus verschiedenen Dateien stammen und zusammengefügt ein Videofilm mit einer Dauer von 9 Sekunden darstellen. Dabei beginnt der Film mit einem roten Coupe, wechselt dann nach drei Sekunden zu einem weißen Coupé, welches in dem Videofilm dargestellt wird und endet mit einer weißen Limousine.

### Patentansprüche

1. Verfahren zum Bereitstellen eines Video-Films, bei dem mindestens ein dargestelltes Objekt während des Ablaufs konfigurierbar ist, umfassend die Verfahrensschritte:

Erstellen einer Vielzahl von Video-Filmen, die verschiedene Konfigurationen des konfigurierbaren Objekts enthalten,

Speichern der Video-Filme unter einem Dateinamen (LIM ROT 01, COU ROT 01, LIM WEI 01, COU WEI 01,) so dass jeder Video-Film in mehrere Fragmente (**9.1.1** bis **9.10.4**) unterteilt ist und

Bereitstellen eines individuellen Video-Films aus einer Folgen von aneinandergehängten Fragmenten (**9.1.2**, **9.2.2**, **9.3.2**, **9.4.4**, **9.5.4**, **9.6.4**, **9.7.4**, **9.8.4**, **9.9.3**, **9.10.3**), wobei immer dann von einem momentan ablaufenden Video-Film mit einer vom Nutzer eines Client-PC (**3**) bestimmten Konfiguration des konfigurierbaren Objekts zu einem anderen Video-Film mit einer anderen Konfiguration des konfigurierbaren Objekts gewechselt wird, wenn die Konfiguration des dargestellten Objekts vom Nutzer des Client-PC (**3**) geändert wird.

2. Verfahren nach Anspruch 1, **dadurch gekennzeichnet**, dass die Fragmente (**9**) der Videofilme Zeitabschnitte mit einer Dauer von weniger als 4 Sekunden (4 s), bevorzugt weniger als 2 Sekunden (2 s) und besonders bevorzugt gleich einer Sekunde (1 s) umfassen.

3. Verfahren nach Anspruch 1 oder 2, **dadurch gekennzeichnet**, dass mindestens ein Fragment (**9**) der Videofilme mit dem Beginn einer Szene des Videofilms anfängt.

4. Verfahren nach einem der Ansprüche 1 bis 3, **dadurch gekennzeichnet**, dass mindestens ein

Fragment (9) der Videofilme mit dem Ende einer Szene des Videofilms endet.

5. Verfahren nach einem der vorhergehenden Ansprüche, **dadurch gekennzeichnet**, dass die Konfiguration des dargestellten Objekts in einem Konfigurationscode enthalten ist.

6. Verfahren nach Anspruch 5, **dadurch gekennzeichnet**, dass der Konfigurationscode mehrere Abschnitte (I, II, III, IV, ...) umfasst.

7. Verfahren nach Anspruch 6, **dadurch gekennzeichnet**, dass diese Abschnitte (I, II, III, IV, ...) des Konfigurationscodes zur Beschreibung der Form (I), der Farbe (II) und/oder der Ausstattung (III) des konfigurierbaren Objekts dienen.

8. Verfahren nach Anspruch 6 oder 7, **dadurch gekennzeichnet**, dass ein Abschnitt (IV) des Konfigurationscodes zur Beschreibung der bit rate des Videofilms dient.

9. Verfahren nach einem der vorhergehenden Ansprüche, **dadurch gekennzeichnet**, dass alle Videofilme die gleiche Spieldauer haben.

10. Verfahren nach einem der vorhergehenden Ansprüche, **dadurch gekennzeichnet**, dass die Bitrate der Videofilme unterschiedlich ist.

11. Computerprogramm für einen Computer, **dadurch gekennzeichnet**, dass es ein Verfahren nach einem der Ansprüche 1 bis 4 sowie 9 und 10 ausführt, wenn es auf einem Computer, insbesondere einem Internet-Server (1) oder einem Content-Delivery-Network (CDN), abläuft.

12. Computer, der als Internet-Server (1) oder als Content-Delivery-Network (CDN) eingesetzt wird, **dadurch gekennzeichnet**, dass er ein Verfahren nach einem der Ansprüche 1 bis 4 sowie 9 und 10 ausführt.

13. Computer, der als Client-PC (3) eingesetzt wird, **dadurch gekennzeichnet**, dass er entsprechend der Eingaben des Benutzers einen Konfigurationscode, bevorzugt nach einem der Verfahrensansprüche 5 bis 8, erzeugt und an einen Internet-Server übermittelt.

Es folgen 3 Seiten Zeichnungen

Anhängende Zeichnungen

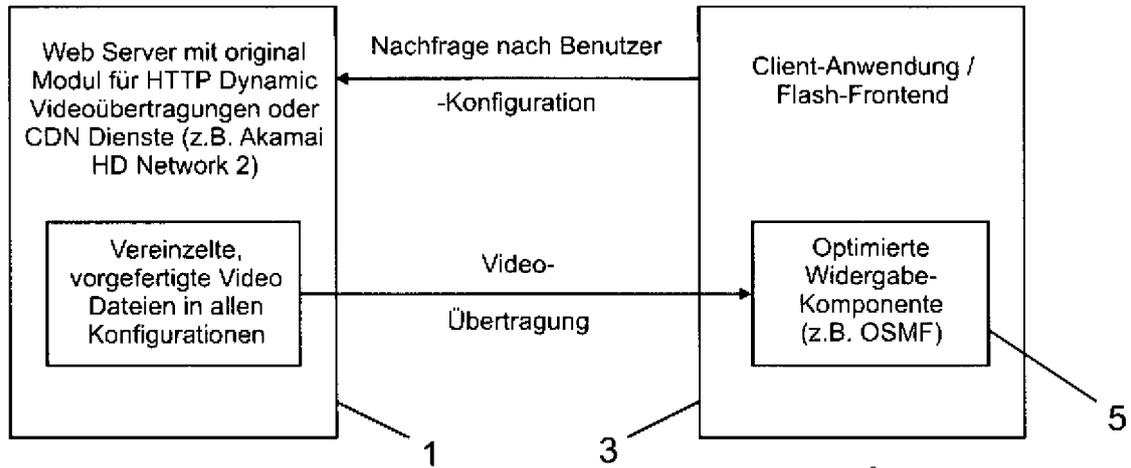


Fig. 1

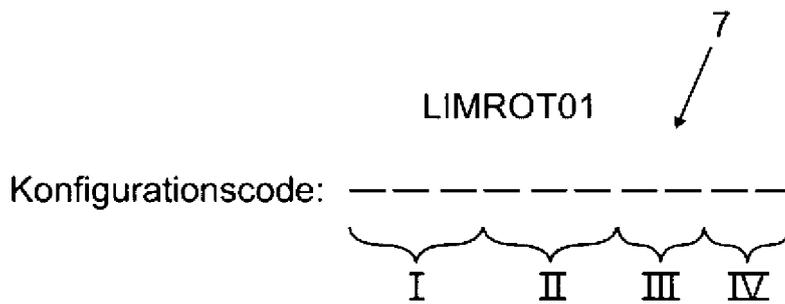


Fig. 2

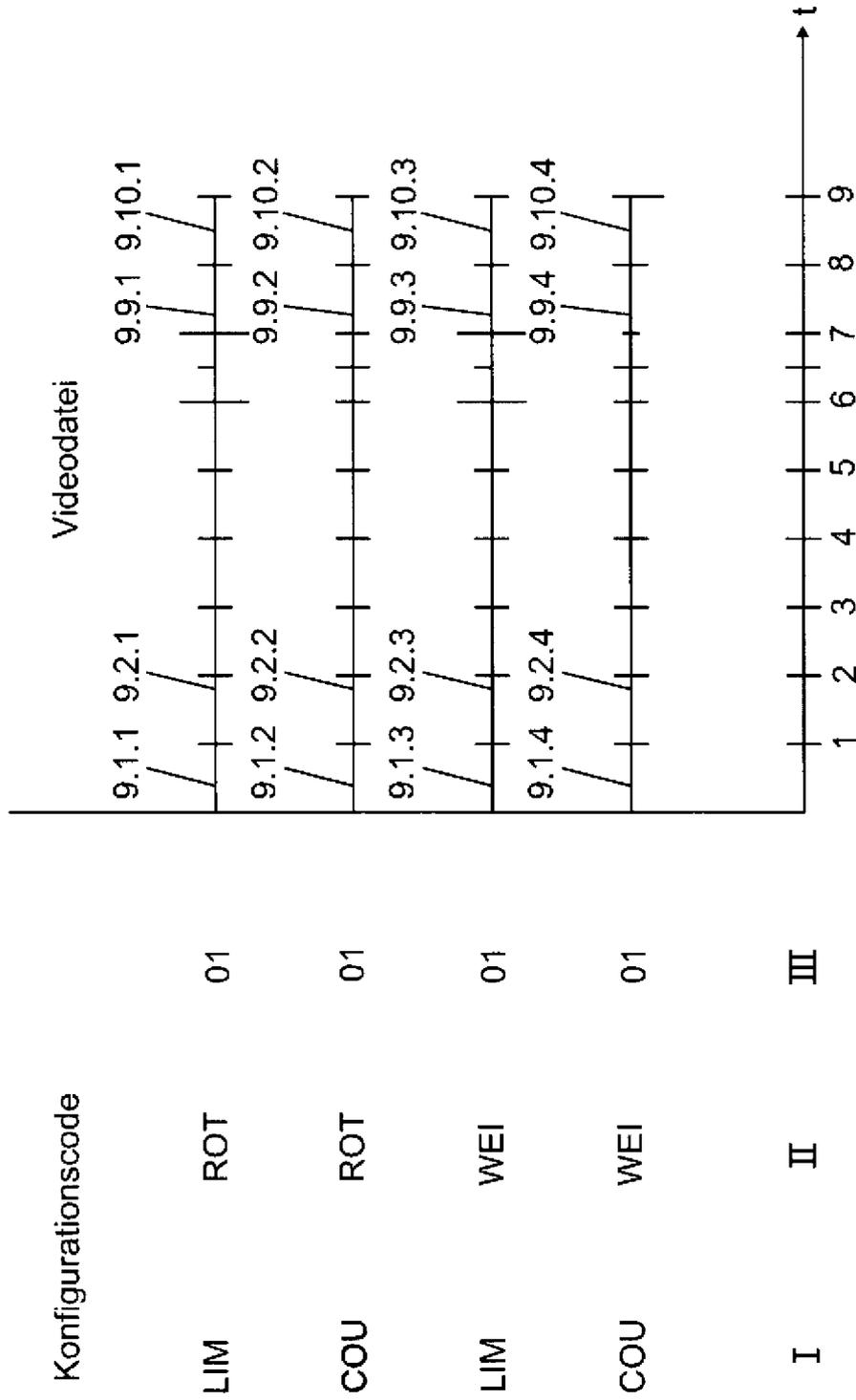


Fig. 3

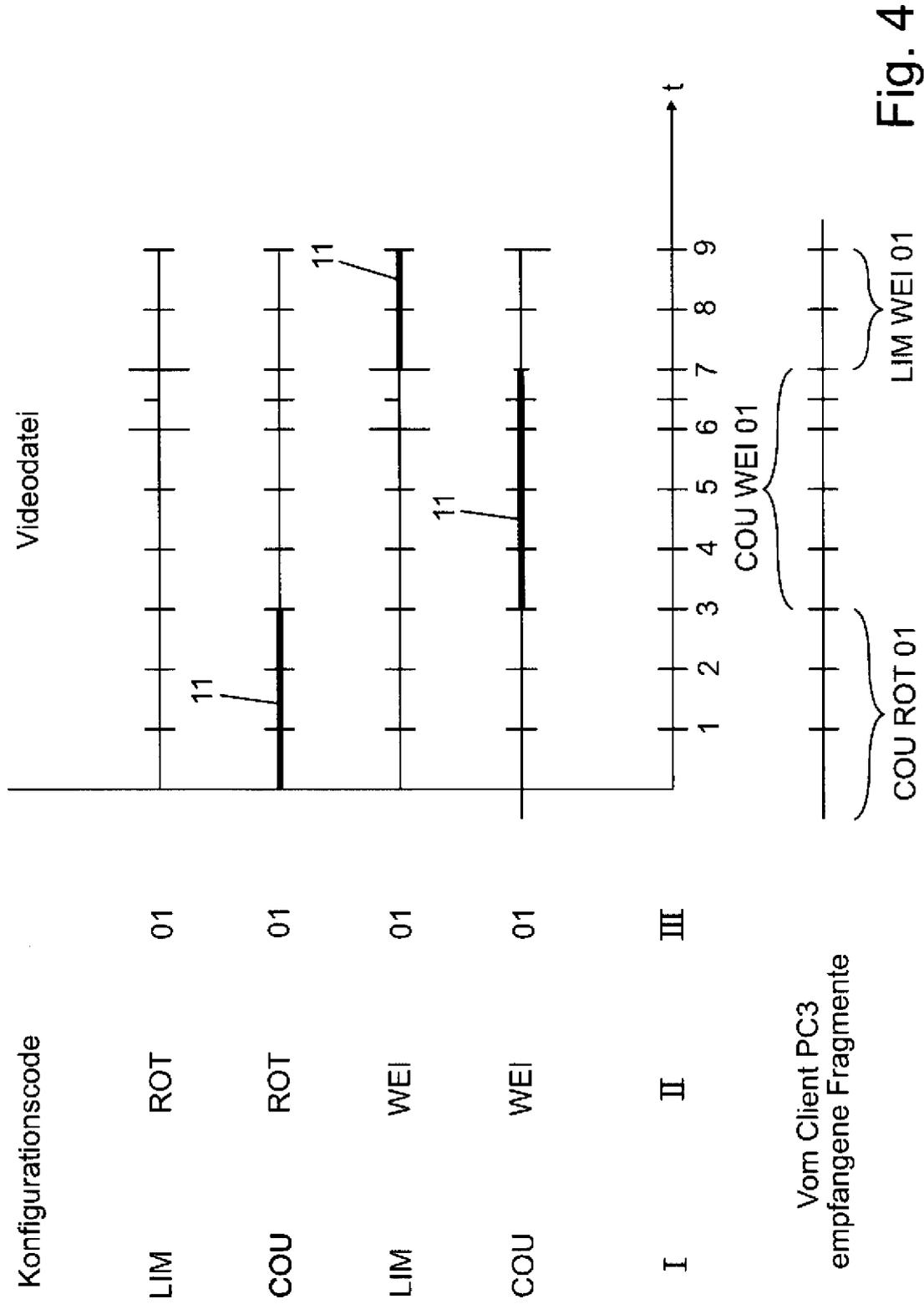


Fig. 4