



(19)
Bundesrepublik Deutschland
Deutsches Patent- und Markenamt

(10) **DE 697 36 298 T2** 2007.05.31

(12) **Übersetzung der europäischen Patentschrift**

(97) **EP 0 958 587 B1**

(21) Deutsches Aktenzeichen: **697 36 298.1**

(86) PCT-Aktenzeichen: **PCT/US97/06005**

(96) Europäisches Aktenzeichen: **97 918 565.9**

(87) PCT-Veröffentlichungs-Nr.: **WO 1997/038434**

(86) PCT-Anmeldetag: **11.04.1997**

(87) Veröffentlichungstag
der PCT-Anmeldung: **16.10.1997**

(97) Erstveröffentlichung durch das EPA: **24.11.1999**

(97) Veröffentlichungstag
der Patenterteilung beim EPA: **05.07.2006**

(47) Veröffentlichungstag im Patentblatt: **31.05.2007**

(51) Int Cl.⁸: **H01J 13/00** (2006.01)
H04L 29/06 (2006.01)

(30) Unionspriorität:
630846 11.04.1996 US

(73) Patentinhaber:
America Online, Inc., Dulles, Va., US

(74) Vertreter:
Henkel, Feiler & Hänzel, 80333 München

(84) Benannte Vertragsstaaten:
DE, FR, GB, IE, SE

(72) Erfinder:
APPELMAN, Barry, McLean, VA 22101, US

(54) Bezeichnung: **REKOMPRIMIERUNGSSERVER**

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).

Die Übersetzung ist gemäß Artikel II § 3 Abs. 1 IntPatÜG 1991 vom Patentinhaber eingereicht worden. Sie wurde vom Deutschen Patent- und Markenamt inhaltlich nicht geprüft.

Beschreibung**HINTERGRUND DER ERFINDUNG****1. Erfindungsgebiet**

[0001] Diese Erfindung bezieht sich auf elektronische Computer und Netzwerksysteme und genauer gesagt auf einen Netzwerksystem zum automatischen Dekomprimieren ausgewählter vorkomprimierter Datenströme und Rekompprimieren der dekomprimierten Daten zu einem größeren Grad als die ursprünglichen vorkomprimierten Daten.

2. Beschreibung des Stands der Technik

[0002] **Fig. 1** ist ein Diagramm eines öffentlichen Weitbereich-Netzwerksystems des Stands der Technik, wie beispielsweise des Internets. Ein erster Knoten **2** (der eine Workstation, ein Terminal, ein Personal-Computer oder dgl. sein kann) kann mit einem zweiten Knoten **4** (der ein Netzwerksystem, ein Mini-computer oder Mainframe-Computer, ein Personal-Computer oder dgl. sein kann) durch eine scheinbar direkte Verbindung **6** kommunizieren. Tatsächlich wird die Verbindung zwischen dem Knoten **2** und dem Knoten **4** durch eine Netzwerk-"Wolke" von Verbindungen **8** hergestellt, im Allgemeinen durch einen oder mehrere "Proxy"-Servercomputer **10**, die helfen, Anforderungen durch das Netzwerk **8** weiterzuleiten. Derartige Proxy-Servercomputer **10** umfassen im Allgemeinen Speichervorrichtungen **12** zum Speichern verschiedener Arten von Daten.

[0003] Jede Verbindung zwischen einem Paar von Knoten **2**, **4** konsumiert etwas von der Kommunikationsbandbreite. Um die Bandbreite zu konservieren, ist es üblich, Dateien vorzukomprimieren, die von einem Knoten **2** zu einem anderen Knoten **4** zu übertragen sind. Beispielsweise können Graphikbilder mit üblicherweise verfügbaren Dateiformaten, wie beispielsweise dem GIF (Graphics Interchange Format), dem TIFF (Tagged Interchange File Format) und dem JPEG vorkomprimiert werden. (Es sollte ersichtlich sein, dass der Begriff "Dateien", wie er hier verwendet wird, Live-Datenströme, Sammlungen oder Archive von Dateien, Abschnitten von Dateien, Datenblöcke etc. umfasst).

[0004] Ein Nachteil des vorliegenden Systems besteht darin, dass derartige vorkomprimierte Dateien nicht optimal komprimiert werden können, womit Bandbreite verschwendet wird. Ein Anforderer-Knoten **2** weist im Allgemeinen jedoch keine Fähigkeit auf, einen Bereitsteller-Knoten **4** zu veranlassen, die Kompression angeforderter Dateien zu optimieren.

[0005] Ein weiteres existierendes Problem besteht darin, dass viele Dateien an Bereitsteller-Knoten **4** überhaupt nicht vorkomprimiert sind. Obwohl eine

Kompression automatisch durch die Verwendung von Transportprotokollen oder Modems, die eine Kompressionsfunktion aufweisen, erreicht werden kann, gibt es keine allgemeine Lösung für dieses Problem.

[0006] Es gibt mehrere Beispiele von Kommunikationsnetzwerken, die in der Technik bekannte Datenkompression benutzt, wobei ein derartiges Beispiel in der EP 616455 beschrieben wird. Bei diesem System werden Nachrichten komprimiert und in Pakete segmentiert, die eine bestimmte Segmentgröße aufweisen, wobei die Kompression mit der gleichen Kompression über alle Verbindungen in dem Netzwerk durchgeführt wird. Die Bandbreite einer bestimmten Verbindung wird maximiert, indem die Segmentgrößen der optimalen Größe für diese Verbindung angepasst wird.

[0007] Ein weiteres Beispiel wird in der US 5131016 beschrieben, die ein Netzwerksystem ist, das beabsichtigt, den höchsten Kompressionsgrad über jede Subverbindung zwischen einem primären Terminal und einem Zielterminal basierend auf der Kompressionsfähigkeit benachbarter stromaufwärtiger und stromabwärtiger Knoten zu erreichen. Insbesondere führt ein primäres Terminal eine Kompressionsverhandlung für eine Verbindung durch, indem die Kompressionsfähigkeit jedes Zwischenknotens und des Zielterminals bestimmt wird. Nach Bestimmen der Kompressionsfähigkeiten berechnet dann das primäre Terminal den maximalen Grad der gemeinsam unterstützbaren Kompression, die zwischen dem primären Terminal und dem Zielterminal zu implementieren ist, um unnötige Kompression und Dekompression zu vermeiden.

[0008] Die Verbindung basiert auf der Erkenntnis des Erfinders eines Bedarfs zum Optimieren der Bandbreitenverwendung in einem Netzwerksystem für vorkomprimierte Dateidateien und eines Bedarfs für ein allgemeines Verfahren zum Komprimieren von Dateien, die nicht vorkomprimiert sind.

[0009] Die Erfindung stellt eine Lösung für diese Probleme bereit.

ZUSAMMENFASSUNG DER ERFINDUNG

[0010] Die Erfindung ist ein Rekomppressionsserver, der automatisch ausgewählte vorkomprimierte Datenströme dekomprimiert und die dekomprimierten Daten zu einem größeren Grad als die ursprünglichen vorkomprimierten Daten rekompprimiert. Bei einer Ausführungsform bestimmt der Rekomppressionsserver aus einer Anforderung, ob eine angeforderte, Datei vorkomprimiert ist. Bei einer weiteren Ausführungsform bestimmt der Rekomppressionsserver aus dem Namen oder Attributen einer wiederhergestellten angeforderten Datei, ob die Datei vorkomprimiert

ist.

[0011] Optional kann der Rekompresseionsserver angeforderte jedoch zuvor unkomprimierte Dateien komprimieren. Als eine weitere Option kann der Rekompresseionsserver häufig angeforderte Dateien in rekomprierter Form zwischenspeichern, um die Bandbreite eines Weitbereichs-Netzwerks weiter zu optimieren. Eine derartige Zwischenspeicherung kann online oder offline durchgeführt werden.

[0012] Die Einzelheiten der bevorzugten Ausführungsform der Erfindung werden in den begleitenden Zeichnungen und der nachstehenden Beschreibung dargelegt. Sobald die Einzelheiten der Erfindung bekannt sind, werden einem Fachmann zahlreiche zusätzliche Neuerungen und Änderungen offensichtlich.

KURZBESCHREIBUNG DER ZEICHNUNGEN

[0013] [Fig. 1](#) ist ein Diagramm eines Weitbereich-Netzwerksystems des Stands der Technik, wie beispielsweise des Internets.

[0014] [Fig. 2](#) ist ein Diagramm der bevorzugten Hardwareausführung der Erfindung.

[0015] [Fig. 3](#) ist ein Blockdiagramm, das den Fluss ausgewählter Daten durch den Rekompresseionsserver von [Fig. 2](#) zeigt.

[0016] [Fig. 4](#) ist ein Ablaufdiagramm, das das Verfahren einer ersten Ausführungsform der Erfindung zeigt.

[0017] [Fig. 5](#) ist ein Ablaufdiagramm, das das Verfahren einer zweiten Ausführungsform der Erfindung zeigt.

[0018] Gleiche Bezugsziffern und Kennzeichnungen in den verschiedenartigen Zeichnungen beziehen sich auf gleiche Elemente.

AUSFÜHRLICHE BESCHREIBUNG DER ERFINDUNG

[0019] Überall in dieser Beschreibung sollten die bevorzugte Ausführungsform und die gezeigten Beispiele als beispielhaft und nicht als Begrenzungen der Erfindung betrachtet werden.

[0020] [Fig. 2](#) ist ein Diagramm der bevorzugten Hardwareausführungsform der Erfindung. Das in [Fig. 2](#) gezeigte Diagramm ist dem in [Fig. 1](#) gezeigten Diagramm ähnlich. Ein Anforderer **2'** (beispielsweise ein Web-Browser auf dem World-Wide-Web-Dienst, der durch das Internet bereitgestellt wird), kann Dateianforderungen an einen Bereitsteller **4'** (beispielsweise einem Web-Server auf dem Internet) über ein

Netzwerk **8** senden. Jede Anforderung wird durch einen Rekompresseionsserver **20** in Übereinstimmung mit der Erfindung verarbeitet.

[0021] Der Rekompresseionsserver **20** umfasst einen Web-Proxy-Server **10'** mit einem optionalen Cache **22**. Der Rekompresseionsserver **20** umfasst ebenfalls einen Datenweg von dem Web-Proxy-Server **10'** entlang einer ersten Verbindung **23** durch einen Dekompressor **24**, entlang einer zweiten Verbindung **25** durch einen Rekompessor **26** und entlang einer dritten Verbindung **27** zurück zu dem Web-Proxy-Server **10'**.

[0022] Die Komponenten entlang des Datenwegs **23-27** können nach Designwahl Hardware, ein Software- oder Firmwareprogrammierter dedizierter Prozessor oder einfach Softwareroutinen sein, die auf dem/den allgemeinen Prozessor(en) des Web-Proxy-Servers **10'** ausgeführt werden.

[0023] [Fig. 3](#) ist ein Blockdiagramm, das den Fluss ausgewählter Daten durch den Rekompresseionsserver von [Fig. 2](#) zeigt. Wenn eine von einem Bereitsteller **4'** angeforderte Datei durch den Rekompresseionsserver **20** läuft, wird eine Bestimmung in dem Web-Proxy-Server **10'** durchgeführt, ob die Datei vorkomprimiert ist. Falls ja, werden die komprimierten Daten **23** entlang einer ersten Verbindung **23** zu einem Dekompressor **24** weitergeleitet. Wenn eine angeforderte Datendatei beispielsweise eine abgebildete Graphik ist, die mit dem GIF-Standard komprimiert ist, gibt die Dateierweiterung (.GIF) an, dass die Datei vorkomprimiert ist. Die Datei würde durch den Dekompressor **24** geleitet werden, der einen herkömmlichen Dekompressionscode zum Dekomprimieren der GIF-Datei aufweist. Die dekomprimierten Daten werden entlang einer zweiten Verbindung **25** (entweder direkt oder nach vorübergehender Speicherung, wie beispielsweise in einem RAM oder einer Massenspeichervorrichtung) zu einem Rekompessor **26** weitergeleitet.

[0024] Der Rekompessor **26** rekompriert die dekomprimierten Daten mit irgendeinem Algorithmus, der ein besseres Kompressionsverhältnis als die ursprüngliche Kompression bereitstellt. Es kann mehrere Rekompresseions-Algorithmen verwendet werden, falls gewünscht, um eine bessere Kompression für unterschiedliche Datentypen bereitzustellen. Bei der bevorzugten Ausführungsform ist der von dem Rekompessor **26** für Graphikdateien verwendete Algorithmus ein "GT" genannter Kompressions-Algorithmus mit veränderbaren Verlust (einschließlich „Verlustfrei“), der von der früheren Johnson-Grace, Inc., Newport Beach, Kalifornien (nun America Online, Virginia) verfügbar ist. Dieser Algorithmus kann abhängig von dem gewünschten Verlustgrad Dateigrößen von 15% (für Null-Verlust) bis 70% kleiner als den GIF-Standard erreichen. Andere Algorithmen

oder Kombinationen von Algorithmen, wie beispielsweise die Huffman-Codierung, die Algorithmen Lempel-Ziv '77, Lempel-Ziv '78 und Lempel-Ziv-Welch für verlustlose Kompression, und MPEG, JPEG, Wavelet und fraktale Algorithmen für verlustreiche Kompression, können verwendet werden.

[0025] Nach Durchlaufen des Rekompessors **26** laufen die rekomprimierten Daten entlang einer dritten Verbindung **27** zu dem Web-Proxy-Server **10'** zur Übertragung an den Anforderer **2'** zurück.

[0026] Als ein optionaler Schritt kann die rekomprimierte Datei in dieser Form in dem Web-Proxy-Server **10'** in einem Massenspeicher-Cache **22** zwischengespeichert werden ([Fig. 2](#)), so dass den Anforderern **2'** für häufig angeforderte vorkomprimierte Dateien eine schnellere Antwort bereitgestellt werden kann. Jeder Cache-Algorithmus kann verwendet werden, wie beispielsweise ein herkömmlicher LRU-Algorithmus ("least recently used" algorithm), um den Cache **22** zu verwalten. Alternativ kann der Web-Proxy-Server **10'** ein Protokoll von Dateianforderungen unterhalten und Dateien auswählen, die basierend auf protokollierten Anforderungshäufigkeiten zu rekomprimieren und zwischenzuspeichern sind. Die Zwischenspeicherung kann online, während des Rekomprimierens von Dateien, oder offline, mit protokollierten Anforderungshäufigkeiten durchgeführt werden, um "populäre" Dateien wieder herzustellen und sie während Leerlaufzeiten für den Web-Proxy-Server **10'** zu rekomprimieren.

[0027] Als eine weitere Option kann der Rekompressionsserver **20** angeforderte, jedoch zuvor unkomprimierte Dateien komprimieren. Dieser Aspekt der Erfindung wird vorteilhafterweise mit der Cache-Option verwendet, da häufig angeforderte unkomprimierte Dateien in dem Rekompressionsserver **20** komprimiert und zwischengespeichert werden können, so dass die Netzwerkbandbreite jedes Mal konserviert wird, wenn eine derartige Datei danach angefordert wird.

[0028] Die Erfindung kann als ein Computerprogramm implementiert werden, das auf einem Medium speicherbar ist, das von einem Computersystem, wie beispielsweise einem Web-Proxy-Server **10'**, gelesen werden kann, um das Computersystem zu konfigurieren, um die hier beschriebenen Funktionen bereitzustellen. Obwohl die Erfindung beschrieben wurde, als ob sie auf einem separaten Prozessor ausgeführt wird, kann sie außerdem als ein Softwareprozess ausgeführt werden, der in dem Web-Proxy-Server **10'** ausgeführt wird.

[0029] [Fig. 4](#) ist ein Ablaufdiagramm, das das Verfahren einer ersten Ausführungsform der Erfindung zeigt. Der Web-Proxy-Server **10'** prüft jede Dateianforderung von einem Anforderer **2'**, um aus der Anforderung selbst zu bestimmen, dass die angeforderte Datei vorkomprimiert ist (beispielsweise durch Prüfen der Dateierweiterung)(Schritt **400**). Wenn die angeforderte Datei nicht vorkomprimiert ist (Schritt **402**), setzt der Web-Proxy-Server **10'** die normale Verarbeitung fort (Schritt **404**).

derung selbst zu bestimmen, dass die angeforderte Datei vorkomprimiert ist (beispielsweise durch Prüfen der Dateierweiterung)(Schritt **400**). Wenn die angeforderte Datei nicht vorkomprimiert ist (Schritt **402**), setzt der Web-Proxy-Server **10'** die normale Verarbeitung fort (Schritt **404**).

[0030] Wenn die angeforderte Datei vorkomprimiert ist (Schritt **402**), wird die Datei auf eine herkömmliche Art und Weise von dem Bereitsteller **4'** wiederhergestellt (Schritt **406**). Die Datei wird dann in dem Dekompressor **24** dekomprimiert (Schritt **408**) und in dem Rekompressor **26** rekomprimiert (Schritt **410**). Die rekomprimierte Datei wird dann an den Anforderer **2'** auf eine herkömmliche Art und Weise weitergeleitet (Schritt **412**).

[0031] Als ein optionaler Schritt kann die rekomprimierte Datei in dieser Form in dem Web-Proxy-Server **10'** in einem Massenspeicher-Cache **22** zwischengespeichert werden (Schritt **414**).

[0032] [Fig. 5](#) ist ein Ablaufdiagramm, das das Verfahren einer zweiten Ausführungsform der Erfindung zeigt. Bei dieser Ausführungsform werden anstatt eine Datei-Anforderung von einem Anforderer **2'** zu prüfen, um zu bestimmen, ob eine Datei vorkomprimiert ist, wie bei dem in [Fig. 4](#) gezeigten Verfahren, Dateien auf herkömmliche Art und Weise wiederhergestellt und dann in dem Web-Proxy-Server **10'** geprüft (Schritt **500**). Ob eine Datei vorkomprimiert ist oder nicht, kann durch die Prüfung der Dateierweiterung (beispielsweise .GIF oder .TIF) oder Schlüsselbytes in einem Dateikopf oder irgendeiner anderen Eigenschaft oder Attribut der Datei, die/das Kompression angibt, bestimmt werden.

[0033] Wenn die angeforderte Datei nicht vorkomprimiert ist (Schritt **502**), setzt der Web-Proxy-Server **10'** die normale Verarbeitung fort (Schritt **504**). Wenn die angeforderte Datei vorkomprimiert ist (Schritt **502**), dann wird die Datei in dem Dekompressor **24** dekomprimiert (Schritt **506**) und in dem Rekompressor **26** rekomprimiert (Schritt **508**). Die rekomprimierte Datei wird dann zu dem Anforderer **2'** auf herkömmliche Art und Weise weitergeleitet (Schritt **510**).

[0034] Als ein optionaler Schritt kann die rekomprimierte Datei in dieser Form in dem Web-Proxy-Server **10'** in einem Massenspeicher-Cache **22** zwischengespeichert werden (Schritt **512**).

[0035] Bei einer alternativen Ausführungsform kann, anstatt jede komprimierte Datei zu rekomprimieren oder jede zuvor unkomprimierte Datei zu komprimieren, zusätzliches Testen durchgeführt werden, um zu entscheiden, ob der oben beschriebene erfinderische Prozess eine Zeitersparnis bei der Übertragung gegenüber der einfachen Neu-Übertragung, einer angeforderten Datei, liefert. Wenn eine

angeforderte komprimierte Datei beispielsweise klein ist, kann die Zeit, die erforderlich ist, um sie zu dekomprimieren und rekomprimieren, die Zeit überschreiten, die erforderlich ist, um sie unverändert zu übertragen. Wenn T_T die geschätzte Übertragungszeit für eine Datei, T_D die geschätzte Dekompressionszeit (Null für eine unkomprimierte Datei) und T_R die geschätzte Rekomppressionszeit (anfängliche Kompressionszeit, wenn die Datei ursprünglich unkomprimiert ist) ist, dann sollte die Erfindung nur verwendet werden, wenn $T_R + T_D < T_T$ ist. Schätzungen für T_T können ohne weiteres durch Messen der tatsächlichen Bitrate zu einem bestimmten Anforderer auf bekannte Art und Weise erhalten werden. Schätzungen für T_R und T_D können erzeugt werden, indem zuerst in einer Vorbereitungsstufe eine statistische Analyse der tatsächlichen Dekompressions- und Rekomppressionszeiten als Funktion der Dateigröße für jeden Kompressionstyp (z.B. GIF, TIFF, JPEG etc.) durchgeführt wird. Somit kann durch Kenntnis der Größe einer bestimmten angeforderten Datei für T_R und T_D eine Schätzung durch Extrapolieren ohne Weiteres ermittelt werden. Alternativ kann ein Schätzwert für T_R und T_D erzeugt werden, indem ein Abschnitt einer Datei dekomprimiert und gleichzeitig versucht wird, diesen Abschnitt zu rekomprimieren, jede Aktion zeitlich gesteuert und auf die vollständige Dateigröße extrapoliert wird.

[0036] In [Fig. 4](#) kann diese Art von Testen vor dem Schritt **408** durchgeführt werden. Wenn der Test angibt, dass $T_R + T_D$ kleiner als T_T ist, dann fährt die Verarbeitung bei Schritt **408** fort. Andernfalls wird die angeforderte Datei direkt zu dem Anforderer weitergeleitet. In [Fig. 5](#) kann diese Art von Testen vor dem Schritt **506** durchgeführt werden. Wenn der Test angibt, dass $T_R + T_D$ kleiner als T_T ist, dann wird die Verarbeitung bei Schritt **506** fortgesetzt. Andernfalls wird die angeforderte Datei direkt zu dem Anforderer weitergeleitet. Bei jedem der beiden Verfahren kann die angeforderte Datei immer noch zwischengespeichert werden.

[0037] Die Erfindung kann in Hardware oder Software oder einer Kombination von beiden implementiert werden. Vorzugsweise wird die Erfindung jedoch in Computerprogrammen implementiert, die auf programmierbaren Computern laufen, die jeweils einen Prozessor, ein Datenspeichersystem (einschließlich flüchtiger und nicht-flüchtiger Arbeitsspeicher- und/oder Speicherelementen), mindestens eine Eingabevorrichtung und mindestens eine Ausgabevorrichtung umfassen. Ein Programmcode wird auf Eingangsdaten angewendet, um die hier beschriebenen Funktionen durchzuführen und Ausgangsinformation zu erzeugen. Die Ausgangsinformation wird auf eine oder mehrere Ausgabevorrichtungen auf bekannte Art und Weise angewendet.

[0038] Jedes Programm ist vorzugsweise in einer

höheren prozeduralen oder objektorientierten Programmiersprache implementiert, um mit einem Computersystem zu kommunizieren. Die Programme können jedoch in Assembler- oder Maschinensprache, falls gewünscht, implementiert sein. In jedem Fall kann die Sprache eine kompilierte oder interpretierte Sprache sein.

[0039] Jedes derartige Computerprogramm wird vorzugsweise auf einem Speichermedium oder einer Speichervorrichtung (z.B. ROM oder Magnetdiskette) gespeichert, die durch einen programmierbaren Allzweck- oder Spezialzweck-Computer lesbar ist, um den Computer zu konfigurieren und zu betreiben, wenn das Speichermedium oder die Speichervorrichtung von dem Computer gelesen wird, um die hier beschriebenen Prozeduren durchzuführen. Das erfindnerische System kann ebenfalls betrachtet werden, als ein computerlesbares Speichermedium implementiert zu sein, das mit einem Computerprogramm konfiguriert ist, wobei das so konfigurierte Speichermedium einen Computer veranlasst, auf eine spezifische und vordefinierte Art und Weise zu arbeiten, um die hier beschriebenen Funktionen durchzuführen.

[0040] Eine Anzahl von Ausführungsformen der Erfindung wurden beschrieben. Nichtsdestotrotz ist es ersichtlich, dass verschiedene Modifikationen ohne Abweichen von dem Schutzbereich der Erfindung durchgeführt werden können. Obwohl die Erfindung im Kontext eines öffentlichen Weitbereich-Netzwerks beschrieben wurde, kann sie beispielsweise auf jedes Netzwerk (einschließlich private Weitbereich- und Lokalbereich-Netzwerke) angewendet werden, bei denen von einem Knoten durch einen anderen Knoten angeforderte Dateien durch einen Zwischenprozessor laufen, der als ein Rekomppressionsserver programmiert oder konfiguriert sein kann. Demgemäß ist ersichtlich, dass die Erfindung nicht durch die spezifische dargestellte Ausführungsform sondern lediglich durch den Schutzbereich der beigefügten Ansprüche begrenzt wird.

Patentansprüche

1. Verfahren zum Erhöhen der verfügbaren Bandbreite in einem Netzwerksystem, mit folgenden Schritten:

- (a) für jede durch einen Anforderer-Knoten (**2'**) von einem Bereitsteller-Knoten (**4'**) in dem Netzwerksystem angeforderte Datei Bestimmen (**402**), ob die Datei eine vorkomprimierte Datei ist, wobei jede Datei eine Mehrzahl von Datenbytes aufweist;
- (b) automatisches Dekomprimieren (**408**) unabhängig von jedem Knoten der gesamten vorkomprimierten Datei in eine verschiedene dekomprimierte Datei;
- (c) automatisches Rekomprimieren (**410**) unabhängig von jedem Knoten der gesamten verschiedenen dekomprimierten Datei in eine verschiedene rekomprimierte Datei mit einer unabhängigen implementier-

ten Kompressionsfähigkeit, wobei die Bytes der verschiedenen rekomprimierten Datei einen größeren Kompressionsgrad bezogen auf die Bytes der vorkomprimierten Datei aufweisen; und
(d) Bereitstellen (**412**) der gesamten rekomprimierten Datei dem Anforderer-Knoten.

2. Verfahren gemäß Anspruch 1, bei dem der Schritt des Bestimmens (**402**), ob eine Datei vorkomprimiert ist, ein Prüfen (**400**) einer Dateianforderung von dem Anforderer-Knoten (**2'**) und dann ein Wiederherstellen (**406**) der vorkomprimierten Datei von dem Bereitsteller-Knoten (**4'**) umfasst.

3. Verfahren gemäß Anspruch 1, bei dem der Schritt des Bestimmens (**502**), ob eine Datei vorkomprimiert ist, ein Prüfen (**500**) einer wiederhergestellten Datei umfasst.

4. Verfahren gemäß Anspruch 1, ferner mit dem Schritt des Zwischenspeicherns (**414**) der rekomprimierten Datei.

5. Verfahren gemäß Anspruch 1, ferner mit folgenden Schritten:

- a. vor dem Schritt des Dekomprimierens (**408**) Bestimmen, ob eine geschätzte Zeit, die erforderlich ist, um die vorkomprimierte Datei direkt dem Anforderer-Knoten (**2'**) bereitzustellen, geringer als eine geschätzte Zeit sein würde, um die vorkomprimierte Datei zu dekomprimieren und die dekomprimierte Datei zu rekomprimieren; und
- b. falls nicht, dann Fortfahren mit dem Schritt des Dekomprimierens, und falls ja, dann Bereitstellen der vorkomprimierten Datei direkt dem Anforderer-Knoten.

6. Vorrichtung (**20**) zum Erhöhen einer verfügbaren Bandbreite in einem Netzwerksystem, mit:

- (a) einem Mittel (**10'**) zum Bestimmen, ob jede durch einen Anforderer-Knoten (**2'**) von einem Bereitsteller-Knoten (**4'**) in dem Netzwerksystem angeforderte Datei eine vorkomprimierte Datei ist, wobei jede Datei eine Mehrzahl von Datenbytes umfasst;
- (b) einem Dekompressor (**24**) zum automatischen Dekomprimieren (**408**) unabhängig von jedem Knoten der gesamten vorkomprimierten Datei in eine verschiedene dekomprimierte Datei;
- (c) einem Rekompessor (**26**) zum automatischen Rekomprimieren (**410**) unabhängig von jedem Knoten der gesamten verschiedenen dekomprimierten Datei in eine verschiedene rekomprimierte Datei mit einer unabhängig implementierten Kompressionsfähigkeit, wobei die Bytes der rekomprimierten Datei einen größeren Kompressionsgrad bezogen auf die Bytes der vorkomprimierten Datei aufweisen;
- (d) einem Mittel (**10'**) zum Bereitstellen der verschiedenen rekomprimierten Datei dem Anforderer-Knoten.

7. Vorrichtung gemäß Anspruch 6, bei der das Mittel (**10'**) zum Bestimmen (**402**), ob eine Datei vorkomprimiert ist, ein Mittel zum Prüfen (**400**) einer Dateianforderung von dem Anforderer-Knoten (**2'**) und dann zum Wiederherstellen (**406**) der vorkomprimierten Datei von dem Bereitsteller-Knoten (**4'**) umfasst.

8. Vorrichtung gemäß Anspruch 6, bei der das Mittel (**10'**) zum Bestimmen (**502**), ob eine Datei vorkomprimiert ist, ein Mittel zum Prüfen (**500**) einer wiederhergestellten Datei umfasst.

9. Vorrichtung gemäß Anspruch 6, ferner mit einem Zwischenspeicher (**22**) zum Speichern der rekomprimierten Datei.

10. Vorrichtung gemäß Anspruch 6, ferner mit einem Prüfmittel zum Bestimmen, ob eine geschätzte Zeit, die erforderlich ist, um die vorkomprimierte Datei direkt an den Anforderer-Knoten (**2'**) zu liefern, geringer als eine geschätzte Zeit sein würde, um die vorkomprimierte Datei zu dekomprimieren (**408**) und die dekomprimierte Datei zu rekomprimieren (**410**), und falls ja, dann Bereitstellen (**412**) der vorkomprimierten Datei direkt dem Anforderer-Knoten.

11. Programm zum Erhöhen der verfügbaren Bandbreite in einem Netzwerksystem, wobei das Computerprogramm auf einem Medium gespeichert ist, das mit einem Computersystem lesbar ist, zum Konfigurieren des Computersystem, wenn es durch das Computersystem gelesen und ausgeführt wird, um die folgenden Funktionen durchzuführen:

- (a) für jede durch einen Anforderer-Knoten (**2'**) von einem Bereitsteller-Knoten (**4'**) in dem Netzwerksystem angeforderte Datei Bestimmen (**402**), ob die Datei eine vorkomprimierte Datei ist, wobei jede Datei eine Mehrzahl von Datenbytes umfasst;
- (b) automatisches Dekomprimieren (**408**) unabhängig von jedem Knoten der gesamten vorkomprimierten Datei in eine verschiedene dekomprimierte Datei;
- (c) automatisches Rekomprimieren (**410**) unabhängig von jedem Knoten der gesamten verschiedenen dekomprimierten Datei in eine verschiedene rekomprimierte Datei mit einer unabhängig implementierten Kompressionsfähigkeit, wobei die Bytes der rekomprimierten Datei einen größeren Kompressionsgrad bezogen auf die Bytes der vorkomprimierten Datei aufweisen;
- (d) Bereitstellen (**412**) der verschiedenen rekomprimierten Datei dem Anforderer-Knoten.

12. Computerprogramm gemäß Anspruch 11, bei dem die Funktion zum Bestimmen (**402**), ob eine Datei vorkomprimiert ist, ein Prüfen (**400**) einer Dateianforderung von dem Anforderer-Knoten (**2'**) und dann ein Wiederherstellen (**406**) der vorkomprimierten Datei von dem Bereitsteller-Knoten (**4'**) umfasst.

13. Computerprogramm gemäß Anspruch 11, bei

dem die Funktion zum Bestimmen (**502**), ob eine Datei vorkomprimiert ist, ein Prüfen (**500**) einer wiederhergestellten Datei umfasst.

14. Computerprogramm gemäß Anspruch 11, ferner mit der Funktion des Zwischenspeichern (**414**) der rekomprimierten Datei.

15. Computerprogramm gemäß Anspruch 11, ferner mit folgenden Funktionen:

(a) vor der Funktion des Dekomprimierens (**408**) Bestimmen, ob eine geschätzte Zeit, die erforderlich ist, um die vorkomprimierte Datei direkt dem Anforderer-Knoten (**2'**) bereitzustellen, geringer als eine geschätzte Zeit sein würde, um die vorkomprimierte Datei zu dekomprimieren (**408**) und die dekomprimierte Datei zu rekomprimieren; und falls nicht, dann Fortfahren mit der Funktion des Dekomprimierens, und falls ja, dann Bereitstellen der vorkomprimierten Datei direkt dem Anforderer-Knoten.

Es folgen 5 Blatt Zeichnungen

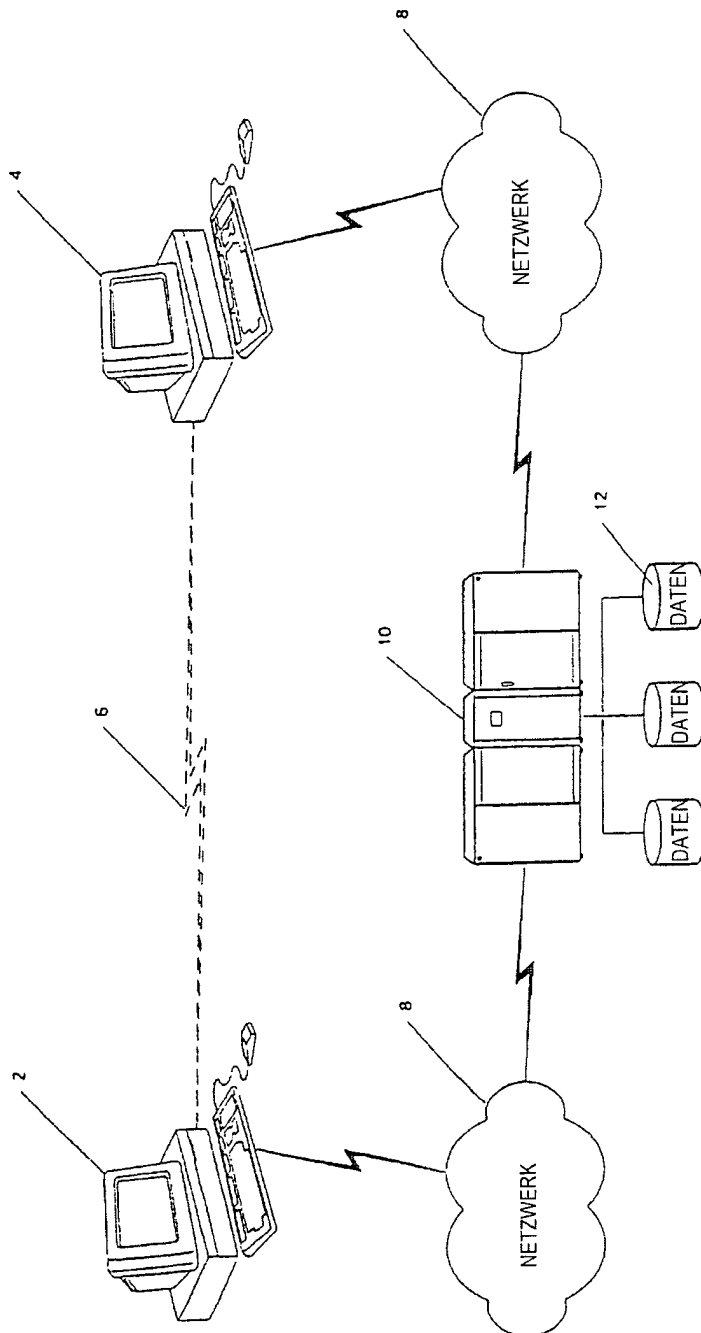


FIG. 1

(STAND DER TECHNIK)

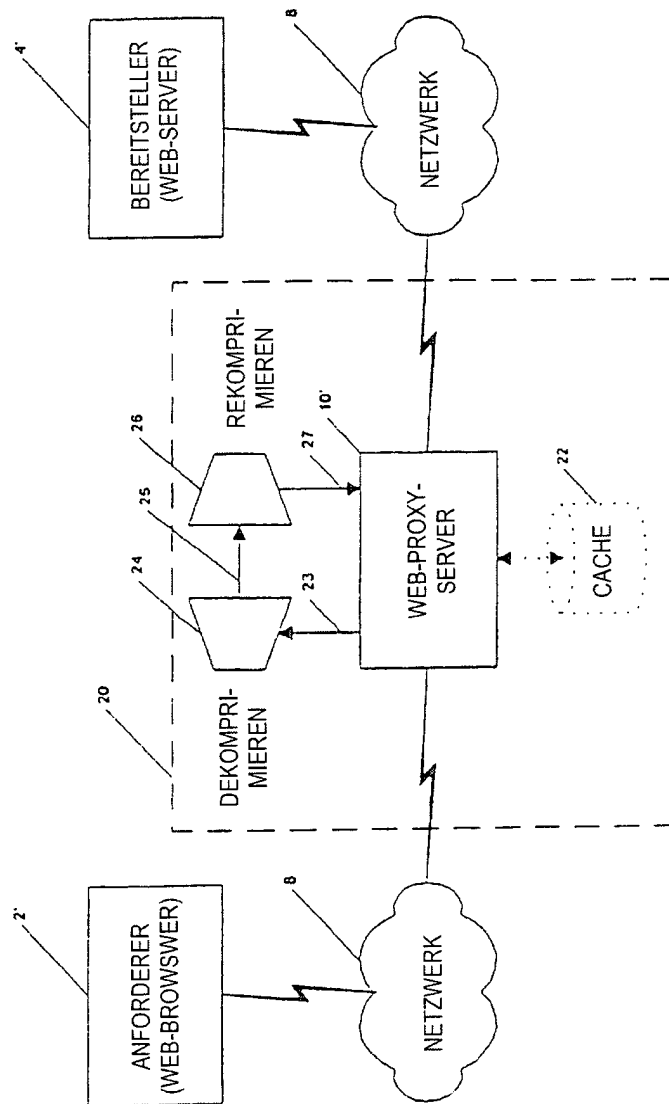


FIG. 2

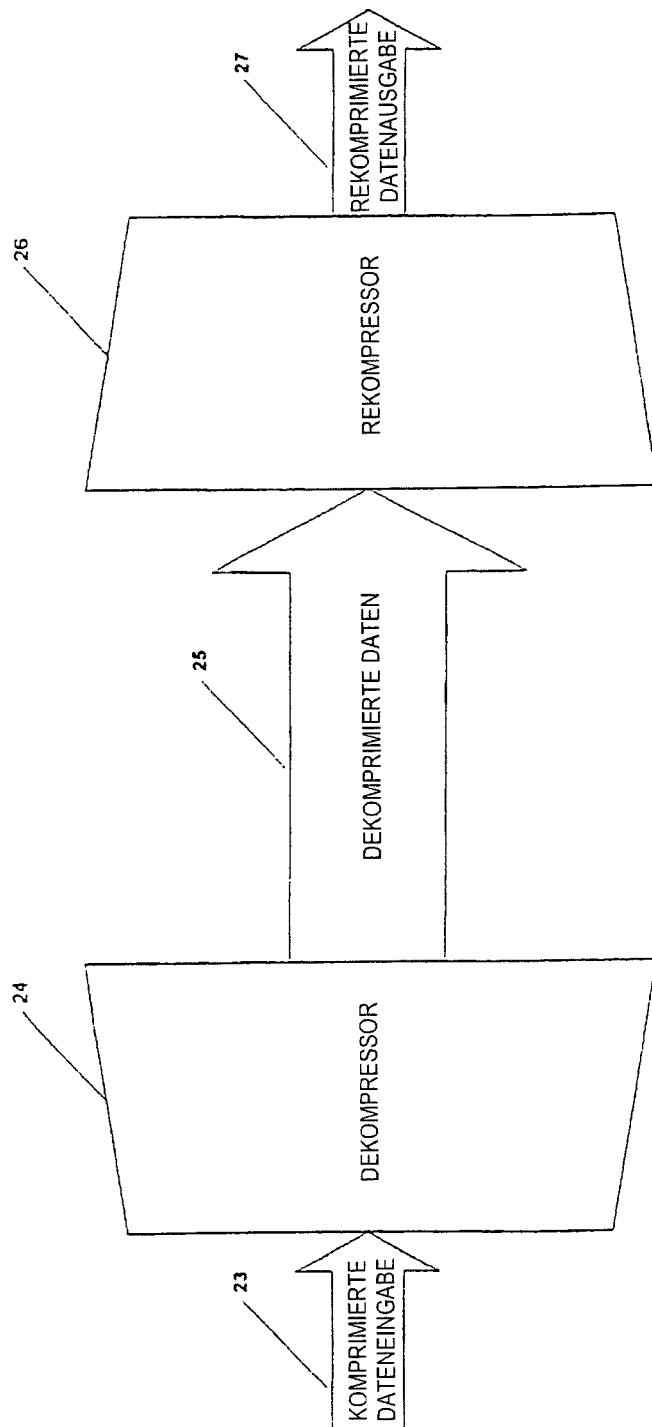


FIG. 3

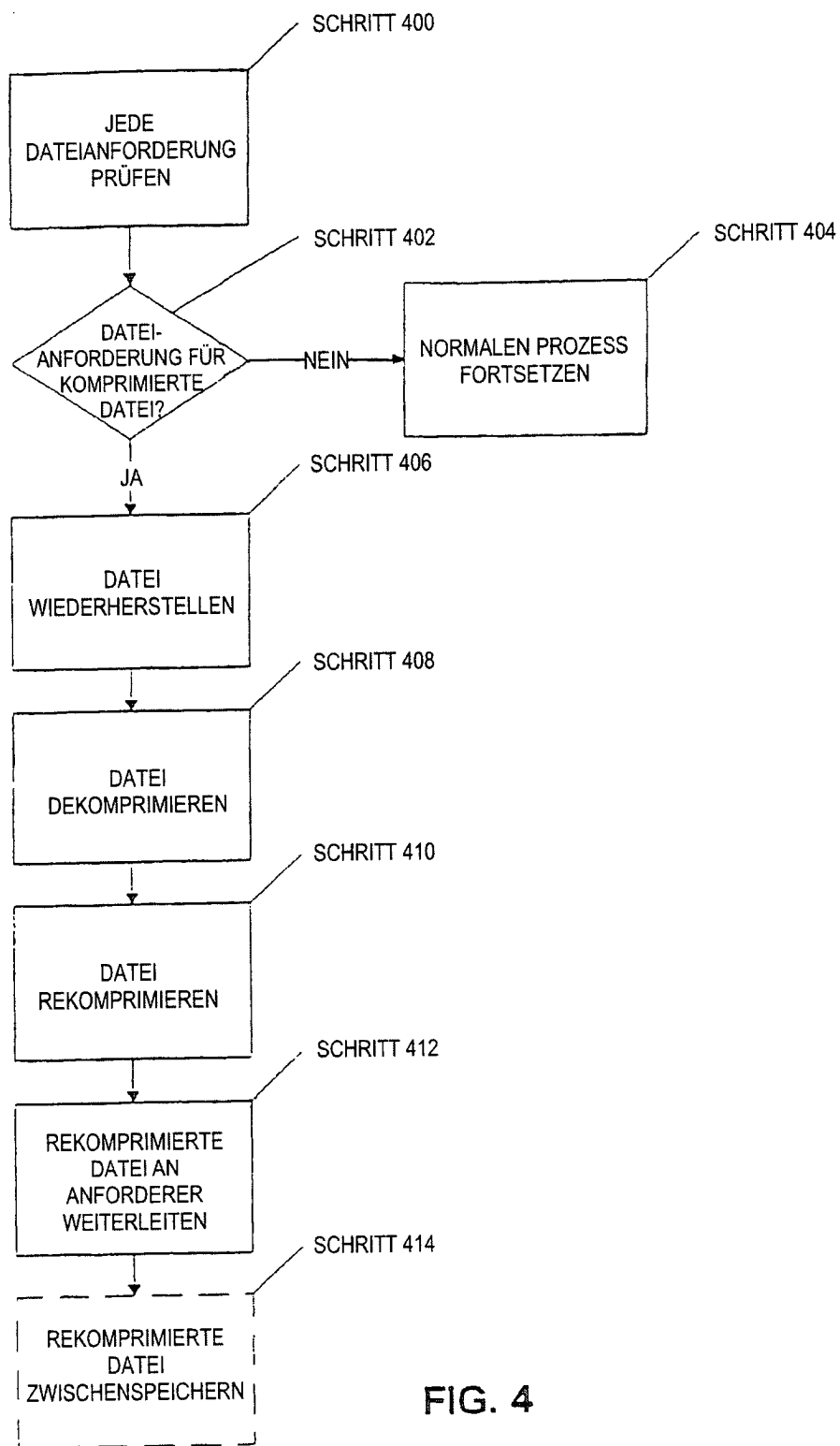


FIG. 4

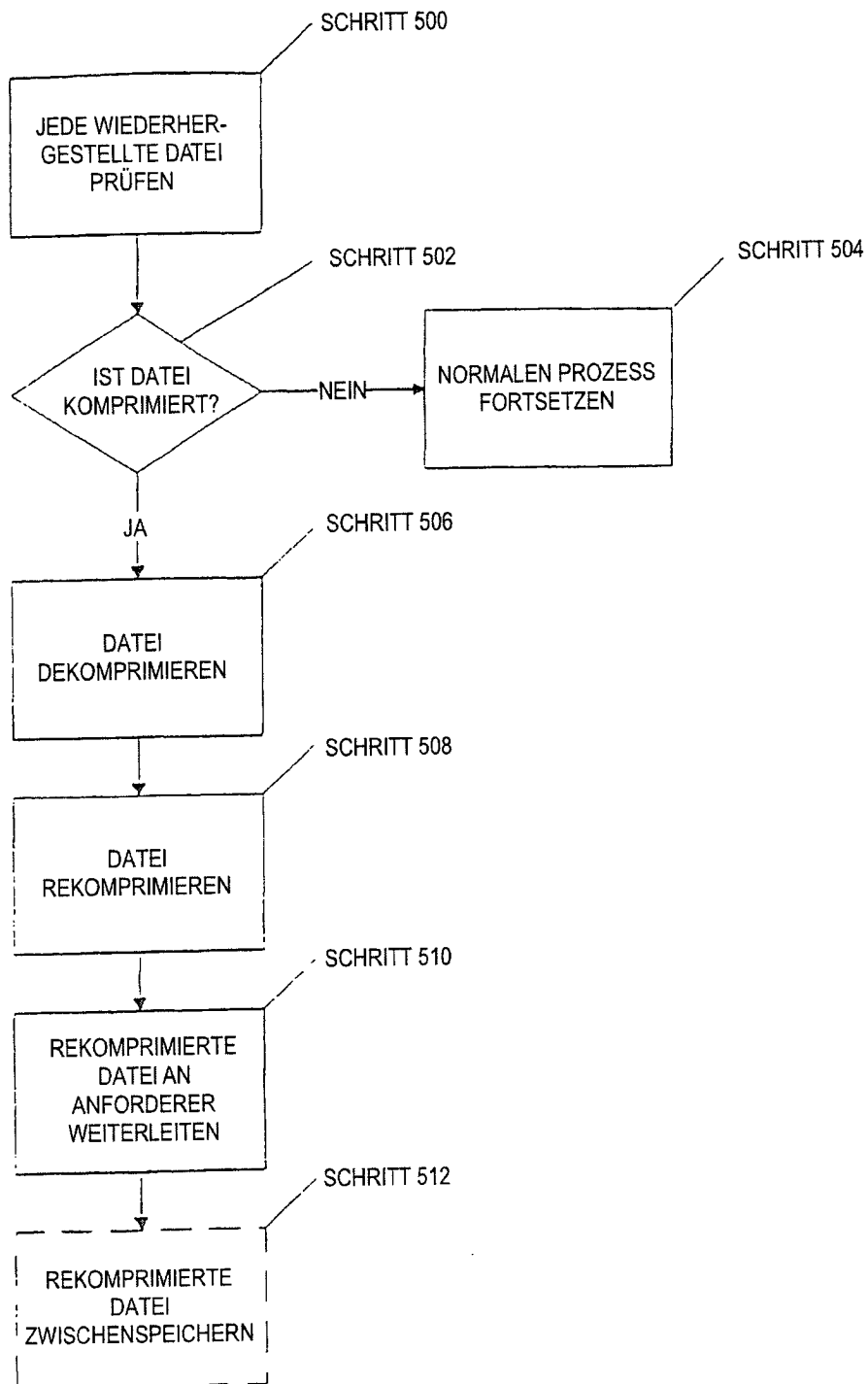


FIG. 5