(19) **Bundesrepublik Deutschland**
Deutsches Patent- und Markenamt

(10) **DE 696 36 201 T2** 2007.05.16

(12) # Übersetzung der europäischen Patentschrift

(97) **EP 1 128 591 B1**
(21) Deutsches Aktenzeichen: **696 36 201.5**
(96) Europäisches Aktenzeichen: **01 112 388.2**
(96) Europäischer Anmeldetag: **16.01.1996**
(97) Erstveröffentlichung durch das EPA: **29.08.2001**
(97) Veröffentlichungstag der Patenterteilung beim EPA: **31.05.2006**
(47) Veröffentlichungstag im Patentblatt: **16.05.2007**

(51) Int Cl.[8]: ***H04L 1/16*** (2006.01)
H04L 29/06 (2006.01)
H04L 12/56 (2006.01)
H04L 1/18 (2006.01)
H04L 12/18 (2006.01)
H04L 5/14 (2006.01)

(30) Unionspriorität:
375493 **19.01.1995** **US**

(84) Benannte Vertragsstaaten:
AT, BE, CH, DE, DK, ES, FR, GB, GR, IE, IT, LI, LU, MC, NL, PT, SE

(73) Patentinhaber:
The Fantastic IP GmbH, Zug, CH

(74) Vertreter:
Hössle Kudlek & Partner, Patentanwälte, 70173 Stuttgart

(72) Erfinder:
Miller, Kenneth C., Concord, MA 01742, US; Robertson, Kary, Newburyport, MA 01950, US; White, Marc, Wayland, MA 01778, US; Cate, Kenneth, Salem, NH 03079, US

(54) Bezeichnung: **Methode zur Mehrfachaussendung in Netzwerken mit ARQ zur Vermeidung unnötiger Wiederholungsübertragungen**

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

Gebiet der Erfindung

[0001] Diese Erfindung betrifft die Datenübertragung und insbesondere schnelle und zuverlässige Multicast-Übertragungen von Dateien von einem Server zu Clients.

Hintergrund der Erfindung

[0002] Computernetzwerke, z. B. weiträumige Netzwerke (WANs), können Unicast-, Multicast- und Broadcast-Dienste bereitstellen, um eine Kommunikation zwischen Netzwerkteilnehmern, z. B. zwischen einem Server-Knoten und einem oder mehreren Client-Knoten, zu ermöglichen. Multicast-Frame-Relay ist ein Dienst, der verwendet wird, um über ein Computernetzwerk zu kommunizieren. Die Multicast-IP-Technologie ist ein weiterer Dienst, der zum Kommunizieren über ein Computernetzwerk verwendet wird. Broadcast-Frame-Relay ist ein Dienst, der verwendet wird, um über ein Satellitennetzwerk zu kommunizieren. Der Begriff "Broadcast" bezieht sich auf einen Server-Knoten, der Informationen zu allen Client-Knoten sendet, die mit dem Netzwerk verbunden sind. Der Begriff "Multicast" bezieht sich auf einen Server-Knoten, der Informationen zu einer Untermenge von allen Client-Knoten sendet, die mit dem Netzwerk verbunden sind. Broadcast und Multicast sind Netzwerkfähigkeiten, die über WANs relativ neu sind.

[0003] Einige Informationsanbieter wollen Informationen elektronisch liefern, indem die Informationen von einem Server-Knoten an einer zentralen Stelle zu einem oder mehreren Client-Knoten an entfernten Kundenstellen über ein Computernetzwerk, mit dem der Server und die Clients gekoppelt sind, durch Broadcast oder Multicast gesendet werden. Da Broadcast- und Multicast-Netzwerkdienste überhaupt keine Bestätigung der gelieferten Informationen vorsehen, können diese Dienste unzuverlässig sein. Eine solche Unzuverlässigkeit ist im Allgemeinen für Informationsanbieter unerwünscht und unannehmbar.

[0004] Eine übliche Protokollreihe zur Verwendung in Computernetzwerken ist TCP/IP, das das im Internet verwendete Protokoll ist. TCP steht für Transmission Control Protocol (Übertragungssteuerprotokoll) und IP steht für Internet-Protokoll. Zwei Dateiübertragungsprotokolle stehen in Zusammenhang mit TCP/IP zur Verfügung: (i) File Transfer Protocol (Dateiübertragungsprotokoll) (FTP), das als Anwendung auf TCP läuft, und (ii) Trivial File Transfer Protocol (triviales Dateiübertragungsprotokoll) (TFTP), das auf UDP läuft. UDP steht für User Datagram Protocol. Sowohl TCP als auch UDP sind Transportprotokolle, die für die Ende-zu-Ende-Lieferung von Informationen über ein Internetzwerk, d. h. ein Netzwerk von Netzwerken, verantwortlich sind.

[0005] Sowohl FTP als auch TFTP unterstützen nur Dateiübertragungen von Punkt zu Punkt (d. h. Unicast). FTP hängt von TCP für eine zuverlässige Lieferung ab, da TCP ein verbindungsorientiertes bestätigtes Transportprotokoll ist. TFTP stellt seine eigenen Bestätigungen für die Zuverlässigkeit bereit, da es auf UDP läuft, das ein verbindungsloser Transportdienst ist, der keine Bestätigung unterstützt.

[0006] Verbindungsorientierte Protokolle wie z. B. TCP erfordern eine Einrichtung und ein Abbrechen von virtuellen Schaltungsverbindungen. Aufgrund ihres relativ hohen Overheads sind TCP und ähnliche Protokolle in Netzwerken mit von Natur aus schlechten Verbindungen, wie z. B. zellenartigen digitalen Paketdaten-Netzwerken (CDPD-Netzwerken) unerwünscht. CDPD verwendet TCP/IP als Hauptprotokollreihe, die im Netzwerk verwendet wird. Drahtlose CDPD-Netzwerke empfehlen Anwendungen, die nur über UDP (die verbindungslose Transportschicht) arbeiten, und folglich ist TFTP das Dateiübertragungsprotokoll der Wahl für CDPD.

[0007] TFTP zerlegt Dateien in Pakete mit jeweils 512 Datenbytes und es sendet dann jedes Datenpaket einzeln. Nachdem jedes Datenpaket gesandt ist, bewirkt TFTP, dass der sendende Knoten auf eine Bestätigung vom (von den) empfangenen Knoten wartet, bevor dem sendenden Knoten gestattet wird, das nächste Datenpaket zu senden. TFTP ist beispielsweise in einem Buch von Douglas E. Corner (Internetworking with TCP/IP, Band I, Principles, Protocols, and Architecture, Zweite Ausgabe, Prentice Hall, 1991, Kapitel 23, Seiten 377-390, beschrieben.

[0008] Obwohl die Bestätigung ein Teil von TFTP ist, wird das bei TFTP verwendete Bestätigungsschema sehr ineffizient, da die Netzwerkverzögerung signifikant wird und/oder für zwei oder mehr der empfangenen Knoten verschieden ist. Wie TFTP erfordern einige andere bekannte Datenübertragungsmechanismen eine paketweise Bestätigung und folglich sind diese anderen Mechanismen auch relativ langsam bei der Übertragung der gesamten Datenmenge.

[0009] "Multicast transport protocols for high speed networks" von Sanjoy et al., Network Protocols, 1994 Proceedings, 1994, Int. Conf. on Boston, MA, USA, 25.-28. Oktober 1994, Los Alamitos, CA, USA, IEEE Comput Soc, offenbart das Multicasting von Datenpaketen zu allen Rezipienten von einer globalen Quelle, das Empfangen von Bestätigungen an der globalen Quelle von einigen der Rezipienten, die als Quellen für lokale Multicast-Bäume wirken, das globale erneute Übertragen von Daten, die von den lokalen Quellen nicht empfangen wurden, das Empfangen von Bestätigungen an den lokalen Quellen von Rezipienten in den lokalen Multicast-Bäumen, das lokale erneute Übertragen von Daten, die von den Rezipienten in den lokalen Multicast-Bäumen von den lokalen Quellen nicht empfangen wurden, und das Bestimmen der Kapazität von einer oder mehreren der Kommunikationsverbindungen, die die Rezipienten mit den Quellen verbinden.

Zusammenfassung der Erfindung

[0010] Gemäß einem ersten Aspekt schafft die vorliegende Erfindung ein Verfahren zum Bestimmen von Kapazitäten oder Framefehlerraten von Kommunikationsverbindungen, wie in Anspruch 1 beansprucht.

[0011] Im Allgemeinen kann die Kommunikationsverbindung, die den Server mit den Clients koppelt und eine Kommunikation zwischen diesen ermöglicht, ein Computernetzwerk (z. B. ein LAN, ein WAN, das Internet), ein drahtloses Netzwerk (z. B. ein zellenförmiges Paketdatennetzwerk wie z. B. CDPD), eine gewisse Kombination dieser Arten von Kommunikationsmedien oder irgendein anderes Kommunikationsmedium, wie beispielsweise ein Satellitennetzwerk, das im Allgemeinen ein Netzwerk mit hoher Geschwindigkeit und hoher Verzögerung ist, sein.

[0012] Gemäß bevorzugten Ausführungsformen der Erfindung senden die Clients nur negative Bestätigungen zum Server zurück, wenn der Server die Dateien sendet. Die Kommunikation ist kontinuierlich. Das heißt, der Server stoppt nicht das Senden der Daten, um auf die negativen Bestätigungen von den Clients zu warten, sondern stattdessen empfängt der Server die negativen Bestätigungen der Clients, wenn der Server die Daten überträgt. Die negativen Bestätigungen der Clients geben dem Server an, welche speziellen Pakete erneut gesandt werden müssen. Ein Paket kann erneut gesandt werden müssen, da es beispielsweise entweder nicht empfangen oder fehlerhaft von einem oder mehreren der Clients empfangen wurde. Nachdem der Server die gesamte Datenmenge (z. B. die gesamte Datei) über die Verbindung zu den Clients gesandt hat, führt der Server eine zweite Runde von Übertragungen durch, in der er nur die speziellen Pakete erneut sendet, die durch die Clients, die eine erneute Übertragung erfordern, angegeben werden. Während dieser zweiten Runde senden die Clients immer noch nur negative Bestätigungen (z. B. Angaben von Paketen, die überhaupt nicht empfangen wurden oder nicht korrekt empfangen wurden). Der Prozess kann dann mit so vielen zusätzlichen Runden von erneuten Übertragungen fortfahren wie erforderlich sind, so dass jeder der Clients alle Pakete korrekt empfängt. Alternativ können die Runden erneuter Übertragung eine vorbestimmte Anzahl von Malen wiederholt werden, wobei die Anzahl modifiziert werden kann (d. h. die Anzahl ist konfigurierbar). Jede anschließende Runde beinhaltet typischerweise die Übertragung von weniger Paketen als die vorherige Runde, da nur vorherige fehlerhafte Pakete erneut gesandt werden.

[0013] Dieses Schema überträgt Daten von einem Server zu einem oder mehreren Clients schnell und zuverlässig. Es ist schnell, da dem Server ermöglicht wird, die gesamte Datei zu übertragen, ohne bei Paketgrenzen zu stoppen, um auf negative Bestätigungen von den Clients für das gerade gesandte Paket zu warten. Das heißt, die Datenübertragung ist insofern nicht direkt mit den negativen Bestätigungen verbunden, als jede Runde der Datenübertragung ungeachtet Empfangsproblemen irgendeines speziellen Clients und/oder ungeachtet irgendwelcher Verbindungsverzögerungsprobleme (z. B. einer Differenz der Zeit, die es Dauert, bis sich ein Paket vom Server zu einem bestimmten Client bewegt, und der Zeit, die es dauert, bis sich ein Paket vom Server zu einem anderen unterschiedlichen Client bewegt) fortfährt. Jede anschließende Übertragungsrunde beinhaltet auch nur das Senden von Paketen, die während der vorherigen Runde nicht empfangen oder fehlerhaft empfangen wurden, und daher muss der Server im Allgemeinen niemals die gesamte Datei mehr als einmal senden. Es ist zuverlässig, da es bestrebt ist, jeden Client mit jedem Paket zu versehen, und die Empfangsprobleme von irgendeinem individuellen Client im Allgemeinen nicht die Empfangsgeschwindigkeit und -genauigkeit der anderen Clients beeinflussen.

[0014] Die Datenübertragung gemäß bevorzugten Ausführungsformen der Erfindung erfordert nicht oder erwartet nicht positive Bestätigungen von irgendeinem der Clients. Eine positive Bestätigung ist impliziert, wenn keine negative Bestätigung am Server zurückempfangen wird.

[0015] Eine Vielzahl von negativen Bestätigungen werden vorzugsweise gesammelt und als "mehrfache selektive negative Abweisungsbestätigung" zum Server zurückgesandt. Typischerweise werden mehr als eine

dieser mehrfachen selektiven negativen Abweisungsbestätigungen beispielsweise während der ersten Runde von Übertragungen vom Server zu den Clients zum Server zurückgesandt. Eine mehrfache selektive negative Abweisungsbestätigung kann Hunderte von einzelnen negativen Bestätigungen darstellen. Die Verwendung dieser Sammlungen von negativen Bestätigungen kann den Verkehr über die Verbindung erheblich verringern und Bandbreite auf der Verbindung für die Übertragung von Daten vom Server zu den Clients und für andere Verwendungen frei machen. Bei der Erfindung werden der Server und die Verbindung im Allgemeinen nicht mit einzelnen negativen Bestätigungen gedrosselt, die alle gleichzeitig oder innerhalb eines kurzen Zeitfensters zurückkommen. Diese Verringerung der Anzahl von einzelnen Bestätigungen, die über die Verbindung zum Server gesandt werden, führt auch zum Nutzen und signifikanten Vorteil von verbesserter Skalierbarkeit. Das heißt, mit der Verwendung der mehreren selektiven negativen Abweisungsbestätigungen, nimmt die Anzahl von Clients, zu denen eine Datei gesandt werden kann, aufgrund des verringerten Bestätigungsverkehrs, der zum Server zurückkommt, zu.

[0016] In einer bevorzugten Ausführungsform der Erfindung wird die gesamte zu übertragende Datenmenge (z. B. eine Datei) in eine Vielzahl von Blöcken aufgetrennt, wobei jeder Block eine Vielzahl von Paketen umfasst. Der Server vollendet eine Runde, wenn er die Übertragung aller Blöcke (z. B. der gesamten Datei) beendet. Nachdem ein vollständiger Block übertragen wurde, senden die Clients ihre negativen Bestätigungen über einen Unicast-Rückführungs-Kommunikationsweg zum Server zurück. Blockgrenzen lösen das Senden von negativen Bestätigungen durch die Clients aus. Wenn die negativen Bestätigungen von den Clients für den Block N in den Server kommen, überträgt der Server den Block N+1 (oder einen anschließenden Block) zu den Clients oder der Server hat die Übertragung aller Blöcke beendet.

[0017] Die folgenden Merkmale werden gemäß bevorzugten Ausführungsformen der Erfindung geschaffen. Es besteht die Fähigkeit, die Übertragungsrate festzulegen und Multicast-Gruppen zu definieren. Es ist auch möglich, die Kapazität von Verbindungen mit unbekannter Kapazität unter Verwendung eines "Multicast-Netzwerksonden"-Merkmals zu bestimmen und die Framefehlerraten von Verbindungen mit bekannter Kapazität unter Verwendung desselben Merkmals zu bestimmen. Ein "Multicast-Ping"-Merkmal kann verwendet werden, um die Konnektivität zwischen einer Quelle und Mitgliedern einer Multicast-Gruppe zu bestimmen. "Geschwindigkeitsgruppen" können nach dem Bestimmen von Verbindungskapazitäten eingerichtet werden, oder wenn sie bereits bekannt sind, wodurch die mit der Quelle durch die schnellsten Verbindungen verbundenen Rezipienten alle Daten empfangen, während Rezipienten mit langsameren Verbindungen nur einen Teil der Daten beim ersten Durchgang empfangen. Die Anzahl von Rezipienten, die die Daten von der Quelle empfangen können, kann erheblich erhöht werden (z. B. um einen Faktor von 1000 oder mehr), indem ein Schema der "Sammlung von negativen Bestätigungen" verwendet wird, wobei "Weiterverbreitungspunkte", vorzugsweise Router, einzelne negative Bestätigungen sammeln und sie als Einheit zur nächsten Ebene weiterleiten.

[0018] Es wird angemerkt, dass die Begriffe "Paket", "Datagramm" und "Frame" hierin austauschbar verwendet werden, um dieselbe Sache zu identifizieren, nämlich eine Einheit von Daten oder Informationen, die eine Quell- und eine Zieladresse als Teil davon besitzen kann und die über die Verbindung gesandt wird.

[0019] Die vorangehenden und weitere Aufgaben, Aspekte, Merkmale und Vorteile der Erfindung werden aus der folgenden Beschreibung und aus den Ansprüchen besser ersichtlich.

Kurzbeschreibung der Zeichnungen

[0020] In den Zeichnungen beziehen sich in den ganzen verschiedenen Ansichten gleiche Bezugszeichen im Allgemeinen auf dieselben Teile. Die Zeichnungen sind auch nicht notwendigerweise maßstäblich, wobei die Betonung stattdessen im Allgemeinen auf die Darstellung der Prinzipien der Erfindung gelegt ist.

[0021] Fig. 1 ist ein Ablaufplan von Datenübertragungsoperationen gemäß bevorzugten Ausführungsformen der Erfindung.

[0022] Fig. 2 ist ein Diagramm einer physikalischen Konfiguration, die einem Server ermöglicht, mit einem oder mehreren Clients zu kommunizieren.

[0023] Fig. 3 ist ein Diagramm, das den Ort einer Ausführungsform der Erfindung in Bezug auf das TCP/IP-Protokollprofil zeigt.

[0024] Fig. 4 ist ein Diagramm eines Blocks eines "ersten Durchgangs" und eines Frame-Übertragungs- und Bestätigungsprozesses.

[0025] Fig. 5 ist ein vereinfachtes Blockdiagramm eines Servers, in dem zumindest ein Teil der vorliegenden Erfindung verkörpert werden kann.

[0026] Fig. 6 ist ein Diagramm eines heterogenen Multicast-Netzwerks mit Mitgliedern einer Multicast-Gruppe, die durch Verbindungen mit unterschiedlicher Kapazität verbunden sind.

[0027] Fig. 7 ist ein Diagramm, das ein Bestätigungssammelmerkmal darstellt, das die Skalierbarkeit erhöht und ermöglicht, dass Millionen von Rezipienten Daten von einem Sender schnell und zuverlässig empfangen.

[0028] Fig. 8 ist ein Diagramm, das eine Stau/Fluss-Steuerung unter Verwendung eines Verfahrens mit variabler Blockgröße betrifft.

[0029] Fig. 9 ist ein Diagramm, das eine Stau/Fluss-Steuerung unter Verwendung eines bevorzugten Statusabfrageverfahrens zum Abfragen von negativen Bestätigungen von Clients vor Blockgrenzen betrifft.

Beschreibung

[0030] Mit Bezug auf Fig. 1 und Fig. 2 umfasst eine schnelle und zuverlässige Datenübertragung von einer Quelle oder einem Server **20** zu einem oder mehreren Rezipienten oder Empfängern oder Clients $\mathbf{22_1}$, $\mathbf{22_2}$, ..., $\mathbf{22_N}$ über eine Kommunikationsverbindung **24** (Schritt **10**) das Übertragen der Daten (z. B. einer Datei), die in Form einer Vielzahl von Frames vorliegen, über die Verbindung **24** zu einem oder mehreren der Rezipienten **22**, bis die gesamte Datei (d. h. alle der Vielzahl von Frames) über die Verbindung **24** übertragen wurde. Wenn die Frames übertragen werden, werden negative Frame-Bestätigungen von einem oder mehreren der Rezipienten **22** über die Verbindung **24** empfangen (Schritt **10**). Wenn, nachdem die gesamte Datei über die Verbindung **24** übertragen wurde, die negativen Bestätigungen angeben, dass bestimmte Frames erneut über die Verbindung **24** übertragen werden müssen (Schritt **12**), werden nur diese bestimmten Frames erneut übertragen (Schritt **14**). Wenn diese bestimmten Frames über die Verbindung **24** erneut übertragen werden, werden negative Frame-Bestätigungen von einem oder mehreren der Rezipienten **22** über die Verbindung **24** empfangen (Schritt **14**). Dieser Prozess wird dann so viele Male wie erforderlich wiederholt, bis keine weiteren Frames erneut übertragen werden müssen, wie durch die Schritte **12**, **14** und **16** angegeben. In Schritt **16** bestimmt der Server **20**, ob "Fertig"-Meldungen am Server **20** von allen Rezipienten **22** empfangen wurden. Wenn ein Rezipient "fertig" ist, bedeutet dies, dass der Rezipient alle Frames empfangen hat und eine "Fertig"-Meldung an den Server **20** gesandt hat, um dies anzugeben. "Fertige" Rezipienten senden weiterhin "Fertig"-Meldungen an den Server, bis sie ihren Namen in einer "Fertigliste" sehen, die der Server als Benachrichtigung zu allen "fertigen" Rezipienten (d. h. den in der "Fertigliste" aufgelisteten) aussendet, um das Senden von "Fertig"-Meldungen an den Server zu stoppen. Nach einem vorbestimmten Zeitraum oder nach einem vorbestimmten Ereignis sendet der Server **20** eine Statusabfrage an alle nicht antwortenden Rezipienten **22**, d. h. Rezipienten, von denen er keine "Fertig"-Meldung empfangen hat (Schritt **18**). Die anfängliche Übertragung der gesamten Datei und jede der anschließenden Übertragungen von Fehlerframes werden hierin im Allgemeinen als "Runde" oder "Durchgang" bezeichnet.

[0031] Im ersten Durchgang sendet der Server **20** vorzugsweise im Multicast-Betrieb die Datei zu einer Untermenge aller Clients **22**. Mindestens zwei der Clients **22** weisen typischerweise eine unterschiedliche diesen zugeordnete Server-Client-Frame-Übertragungsverzögerung auf. Die Datenübertragung gemäß der Erfindung ist durch solche Verzögerungsdifferenzen unbeeinflusst, selbst wenn signifikante und sogar wenn jeder Client **22** eine andere diesem zugeordnete Verzögerung besitzt.

[0032] Die Verbindung **24** kann ein Computernetzwerk (z. B. ein LAN, ein WAN, das Internet), ein drahtloses Netzwerk (z. B. ein zellenförmiges Datennetzwerk), eine gewisse Kombination dieser zwei Arten von Kommunikationsmedien oder irgendein anderes Kommunikationsmedium, wie beispielsweise ein Satellitennetzwerk, das typischerweise eine hohe Geschwindigkeit und hohe Verzögerung aufweist, sein. Die Vielzahl von Frames, die über die Verbindung **24** während der ersten Runde übertragen werden, kann zusammen eine Computerdatei darstellen, die vom Server **20** zu einem oder mehreren der Clients **22** übertragen wird.

[0033] Der Server **20** und die Clients **22** können Computer wie z. B. PCs oder Arbeitsplatzrechner sein, die auf irgendeinem einer Vielzahl von Betriebssystemen laufen, einschließlich DOS. Mit Bezug auf Fig. 5 umfasst der Server **20** ungeachtet dessen, um welche Art von Computer es sich handelt, typischerweise einen zentralen Prozessor **50**, eine Hauptspeichereinheit **52** zum Speichern von Programmen und/oder Daten, eine Eingabe/Ausgabe-Steuereinheit **54**, eine Netzwerkschnittstelle **56**, eine oder mehrere Eingabevorrichtungen **58** wie z. B. eine Tastatur und eine Maus, eine Anzeigevorrichtung **60**, eine feste oder Festplattenlaufwerkseinheit **62**,

eine Diskettenlaufwerkseinheit **64**, eine Bandlaufwerkseinheit **66** und einen Datenbus **68**, der diese Komponenten koppelt, um eine Kommunikation zwischen diesen zu ermöglichen. Jeder der Client-Computer **22** umfasst im Allgemeinen alle oder einige der im Server **20** von Fig. 5 enthaltenen Komponenten.

[0034] In einigen Ausführungsformen definieren ein oder mehrere Computerprogramme die Betriebsfähigkeiten des Servers **20** und der Clients **22**. Die Programme können in den Server **20** und die Clients **22** über das Festplattenlaufwerk **62**, das Diskettenlaufwerk **64** und/oder das Bandlaufwerk **66** geladen werden. Alternativ können sich die Programme in einem permanenten Speicherabschnitt (z. B. einem ROM-Chip) des Hauptspeichers **52** befinden. In einigen anderen Ausführungsformen können der Server **20** und/oder die Clients **22** speziell ausgelegte, zweckgebundene festverdrahtete elektronische Schaltungen umfassen, die alle hierin beschriebenen Funktionen ohne den Bedarf für Befehle von Computerprogrammen durchführen. Die Erfindung kann beispielsweise verwendet werden, um neue Revisionsebenen der Client-Software elektronisch vom Server auf einen oder mehrere der Clients schnell und zuverlässig zu laden.

[0035] Mit Bezug auf Fig. 3 arbeitet die Erfindung vorzugsweise auf der Anwendungsschicht **30** des TCP/IP-Protokollprofils **32** auf UDP. Die Erfindung könnte auch auf der Anwendungsschicht über der verbindungslosen Transportschicht, die in anderen Protokollprofilen wie z. B. IPX in der NetWare SPX/IPX-Protokollreihe vorliegt, arbeiten. UDP steht für User Datagram Protocol und es ist das TCP/IP-Standardprotokoll, das ermöglicht, dass ein Anwendungsprogramm auf einem Computer ein Datagramm zu einem Anwendungsprogramm auf einem anderen Computer sendet. UDP verwendet das Internet-Protokoll (IP), um Datagramme zu liefern. UDP-Datagramme unterscheiden sich von IP-Datagrammen darin, dass UDP-Datagramme eine Protokoll-Portnummer umfassen, die dem Sender des Datagramms ermöglicht, unter mehreren Zielorten (d. h. Anwendungsprogrammen) auf dem empfangenden Computer zu unterscheiden. UDP-Datagramme umfassen typischerweise auch eine Prüfsumme für die gesandten Daten.

[0036] Im Allgemeinen umfasst die Datenübertragung gemäß der Erfindung vier Aspekte: LEERLAUF, ANKÜNDIGUNG/REGISTRIERUNG, ÜBERTRAGUNG und VOLLENDUNG. Im LEERLAUF-Zustand besteht keine Aktivität. Wenn eine Sammlung von Daten (z. B. eine Datei) für die Übertragung durch den Server **20** ausgewählt wird, wird in die ANKÜNDIGUNGS/REGISTRIERUNGS-Phase eingetreten. Während irgendeiner der vier Phasen stehen alle Dateien für einen Operator am Server **20** zur Verfügung.

ANKÜNDIGUNG/REGISTRIERUNG

[0037] In dieser Phase (Schritt **8** in Fig. 1) kündigt der Server den Clients an, dass eine Datei gleich übertragen wird, und liefert die der Übertragung der Datei zugeordneten Parameter. Die maximale Dauer dieser Phase wird in Minuten ausgedrückt und ist konfigurierbar. Eine ANKÜNDIGUNGS-Meldung wird verwendet, um Multicast-Gruppen einzurichten, und Adressen der Klasse D werden bei der Zuweisung von Multicast-Gruppen verwendet.

[0038] Clients sind verpflichtet, beim Server zu registrieren, dass sie eine ANKÜNDIGUNGS-Meldung empfangen haben. Wenn ein Client die ANKÜNDIGUNGS-Meldung sieht, überprüft der Client, ob sie der in der Meldung identifizierten Gruppe zugeordnet ist. Es ist im Empfänger, der die ANKÜNDIGUNGS-Meldung verarbeiten kann, impliziert, dass der Empfänger eine korrekte Server-IP-Adresse und eine korrekte Portnummer besitzt. Die Clients antworten automatisch auf ANKÜNDIGUNGS-Pakete mit REGISTRIERUNGS-Paketen, bis sie ihre Adresse in einer Liste registrierter Clients in einem nachfolgenden ANKÜNDIGUNGS-Paket sehen. Das REGISTRIERUNGS-Paket wirkt als positive Bestätigung für den Server über die Teilnahme des Clients. Sobald der Server das REGISTRIERUNGS-Paket des Clients empfängt, fügt der Server den Client zur Client-Liste beim nächsten Broadcast des ANKÜNDIGUNGS-Pakets hinzu. Die Client-Liste wird vom Server gewartet. Wenn der Client ein ANKÜNDIGUNGS-Paket mit der ID des Client in der Client-Liste empfängt, ist die Registrierung für den Client vollständig. Wenn alle erwarteten Empfänger auf die ANKÜNDIGUNGS-Meldung geantwortet haben oder der ANKÜNDIGUNGS-Zeitablauf abgelaufen ist, was auch immer zuerst geschieht, beginnt die eigentliche Übertragung der Datei. Diese Registrierung gibt an, dass der Client an der Gruppe teilnehmen kann, da er die Ressourcen zum Handhaben der gleich zu sendenden Datei besitzt. Um eine ungewollte Teilnahme zu verhindern, kann ein Verschlüsselungsschlüsselaustausch bei der Gruppenaufstellung stattfinden. Sobald die Dateiübertragung beginnt, hört das Senden von ANKÜNDIGUNGS-Paketen auf und die ANKÜNDIGUNGS-Phase ist vorüber (Schritt **9** in Fig. 1).

[0039] Alle Eigenschaften der Dateiübertragung werden im ANKÜNDIGUNGS-Paket übertragen. Beim Empfang dieser ANKÜNDIGUNGS-Meldung antwortet der Client dem Server mit einem Unicast-Datagramm. Die Antwort gibt an, ob der Empfänger die Einrichtungen zum Empfangen der Datei besitzt oder nicht. Sie gibt im

Fall einer abgebrochenen Übertragung auch an, ob der Client genügend Kontext besitzt, um die Übertragung wieder aufzunehmen (ein "Neustart", wie in Fig. 1 angegeben). Die Dauer der Ankündigungsperiode sollte in einigen Fällen ermöglichen, dass ein Operator an der Serverstelle einen Aufruf an die Client-Stelle einleitet, der angibt, dass der Computer entweder nicht erreichbar ist oder nicht die Einrichtungen für die Übertragung besitzt. An der Client-Stelle könnten die Korrekturen entweder manuell oder, falls so konfiguriert, unter Fernsteuerung vom Server vorgenommen werden, um Ressourcen frei zu machen, so dass er an der Übertragung teilnehmen kann.

[0040] Zu irgendeinem Zeitpunkt während der ganzen Übertragung kann der Client auf dieses Paket antworten, wobei er angibt, dass er die Übertragung von seinem Ende abgebrochen hat, wobei er den Grund in der Meldung angibt. Wenn eine Übertragung vor der Vollendung abgebrochen wird, kann die Erfindung später diese wieder aufnehmen, ohne Teile der Datei, die bereits erfolgreich gesandt wurden, erneut zu senden "Neustart" in Fig. 1). Dies ist ein besonders wichtiges und nützliches Merkmal, wenn sehr große Dateien gesandt werden. Um dieses Merkmal zu erreichen, verwirft ein Client keine teilweise empfangene Datei. Stattdessen speichern die Clients teilweise empfangene Dateien. Wenn Probleme bestehen, die verhindern, dass alle Clients (z. B. alle Clients in einer Multicast-Gruppe) die ganze Datei empfangen, wenn sie zum ersten Mal gesandt wird (z. B. die Verbindung aus irgendeinem Grund während der Dateiübertragung geschlossen wird), kann die Übertragung später erneut gestartet werden, um die Übertragung zu vollenden. Während eines Neustarts fragt der Server alle Clients für eine Liste von Datenframes, die verpasst wurden, ab und dann beginnt der Server die Vollendung der Übertragung durch Senden nur dieser Frames. In Fig. 1 beinhaltet Schritt **10** folglich für einen Neustart eine Übertragung, die zuerst mit den Frames startet, die während der anfänglichen abgebrochenen Übertragung verpasst wurden (d. h. Nak'd), anstatt mit dem ersten Frame des ersten Blocks der Datei zu starten, wie es bei einem unabgebrochenen normalen Start einer Übertragung geschehen würde.

ÜBERTRAGUNG

[0041] Beim Eintreten in die Datenübertragungsphase wird ein Übertragungsprotokoll am Server gewartet. Dieses Protokoll ist immer aktiv und es verfolgt alle Ereignisse. Jeder der Clients wartet auch ein Übertragungsprotokoll. Das an jedem der Clients gewartete Protokoll wird nachstehend unter der Überschrift "VOLLENDUNG" erwähnt.

[0042] Da Dateien mit 2 Gigabytes von Daten oder mehr übertragen werden können, ist das Halten der gesamten Datei im Speicher am Server für das Ausmaß der Übertragung im Allgemeinen unrealistisch. Die Anzahl von Clients, die die Datei empfangen sollen, kann 1000 oder mehr sein, und folglich ist das Anhalten der Übertragung, um auf Bestätigungen von jedem von ihnen zu warten, bevor mit der nächsten Blockübertragung fortgefahren wird, unannehmbar.

[0043] Der Server unterteilt jede zu übertragene Datei logisch in Blöcke von Frames und jeder Block umfasst typischerweise eine Vielzahl von Frames und möglicherweise Tausende von Frames. Mit Bezug auf Fig. 4 hat der Server **20** in einem Beispiel eine Datei in vier Blöcke unterteilt, nämlich Block 1, Block 2, Block 3 und Block 4, wobei jeder Block einen oder mehrere Frames umfasst. Jeder Block stellt eine Einheit dar, die von jedem Client, der an einer Übertragung teilnimmt, negativ bestätigt wird (nur keine positiven Bestätigungen), wenn der Client feststellt, dass ein Block durch den Server gesandt wurde. Der Client erfasst dies durch eine Änderung der Blocknummer in empfangenen Datenpaketen, da jeder gesandte Frame seine Blocknummer und seine Frame-Nummer innerhalb dieses Blocks angibt. Das Aufteilen der Datei in Blöcke stellt mindestens zwei Vorteile bereit: (i) Verringern der Anzahl von erforderlichen negativen Bestätigungen; und (ii) Verringern der Speicheranforderungen im Server zum Bestimmen von nächsten Dateidurchgangs-Übertragungsblöcken.

[0044] Datenübertragungen sind nicht direkt mit den negativen Bestätigungen verbunden. Die Übertragung fährt ungeachtet von verpassten negativen Bestätigungen oder vorher verpassten Datenpaketen durch irgendeinen einzelnen Client fort. Dies ermöglicht eine Einfachheit der Konstruktion und stellt sicher, dass Probleme einzelner Clients eine minimale Auswirkung auf die Gruppe als Ganzes schaffen. Es ist auch zu beachten, dass Clients für das Senden von negativen Blockbestätigungen auf der Basis dessen, was sie vom Server hören, verantwortlich sind.

[0045] Mit Bezug auf Fig. 4 startet der Server die Übertragung durch Senden des ersten Frames des ersten Blocks (d. h. des ersten Frames von $Block_1$). Dieser Server sendet die Frames mit einer Rate, die konfigurierbar ist. Dies stellt die Basisübertragungsrate dar, die auf der Basis der Leistung zurückgedrosselt (d. h. verringert) werden kann. Der Server setzt das Senden der Frames der Datei fort, bis die vollständige Datei einmal in das Netzwerk gesandt wurde (d. h. bis der $Block_1$ bis $Block_4$ gesandt sind). Dies ist als erster Durchgang

oder erste Runde definiert und es dauert eine Menge an Zeit, die in Fig. 4 als "B_4" dargestellt ist. Einige Clients können die vollständige Datei (d. h. alle vier Blöcke) nach dem ersten Durchgang korrekt empfangen, in welchem Fall sie den Empfang der Datei beendet haben. Clients, die einen oder mehrere fehlerhafte Frames empfangen oder überhaupt keinen oder keine Frames empfangen, erfordern das erneute Senden von bestimmten "Stücken" der Datei (d. h. die fehlerhaft empfangenen oder verpassten Frames) in nachfolgenden Durchgängen oder Runden. Jeder nachfolgende Durchgang oder jede nachfolgende Runde erfordert die Übertragung von weniger Frames, da nur negativ bestätigte Frames (d. h. Frames, die nicht empfangen oder fehlerhaft empfangen wurden) in der vorherigen Runde in der nachfolgenden Runde erneut übertragen werden.

[0046] Ein maximaler Durchgangszählwert oder eine maximale Zeit zum Vollenden kann ein konfigurierbarer Parameter sein. Es kann Clients geben, die bis zu einer maximalen Durchgangs- oder einer maximalen Zeitdauer nicht die ganze Datei korrekt empfangen haben. Diese Clients werden vom Server identifiziert und der Server kann eine weitere Handlung unternehmen, um diesen Clients den Rest der Informationen über beispielsweise einen Unicast-Dateiübertragungsprozess zu beschaffen. In der bevorzugten Ausführungsform senden die Clients "Fertigmeldungen", die angeben, dass sie die ganze Datei empfangen haben, und der Server sendet "Fertiglisten", die Clients angeben, von denen behauptet wird, dass sie "fertig" sind. Wenn nach einem vorbestimmten Ereignis (z. B. einer vorbestimmten Menge an Zeit) der Server keine "Fertigmeldung" von bestimmten Clients empfängt und alle NAKs bedient wurden, sendet der Server eine Statusabfragemeldung zu diesen Clients und sendet irgendwelche fehlenden Frames zu Clients, die mehr Daten benötigen. Irgendeinem Client, der immer noch nicht antwortet, kann die Datei beispielsweise in einer Unicast-Übertragung zu diesem Client vom Server zu einer späteren Zeit gesandt werden.

[0047] Wenn der Server die Blockgrenzen (d. h. B_1, B_2, B_3 und B_4 in Fig. 4) überschreitet, senden die einzelnen Clients vorzugsweise "mehrfache selektive negative Abweisungs"-Bestätigungen ("Nak") für jeden Block. Diese Bestätigungen von den Clients für jeden Block werden am Server eine gewisse Zeit, nachdem die Grenze dieses Blocks überschritten wird, empfangen. Positive Bestätigungen sind impliziert. Eine mehrfache selektive negative Abweisungsbestätigung für einen speziellen Block bedeutet, dass ein oder mehrere Frames in diesem speziellen Block von denjenigen Clients, die angeben, dass das Netzwerk sie aus irgendeinem Grund nicht geliefert hat, fehlerhaft empfangen wurden oder überhaupt nicht empfangen wurden. Folglich geben zum Server gesandte Bestätigungen an, welche Frames fehlerhaft empfangen oder nicht empfangen wurden.

[0048] Bei nachfolgenden Durchgängen (d. h. nach dem in Fig. 4 gezeigten ersten Durchgang) antworten Clients nur mit negativen Bestätigungen für Blöcke, die wieder nicht korrekt empfangen wurden. Da der Server Stücke (Frames) der Datei, die von verschiedenen Clients benötigt werden, zu allen Clients in nachfolgenden Durchgängen sendet, haben viele der Clients sie bereits korrekt beim ersten Durchgang empfangen und ignorieren sie folglich.

[0049] Im Allgemeinen können alle Informationen, die von den Clients zum Server zurückkehren, auf einem Rückweg übertragen werden, der von dem (den) Weg(en) getrennt ist, den/die der Server verwendet, um die Frames zu den Clients zu übertragen. Für die Zwecke dieser Beschreibung sollte jedoch die Kommunikationsverbindung **24** (Fig. 2) oder ein anderer Weg, der ermöglicht, dass der Server und die Clients kommunizieren, im Allgemeinen so aufgefasst werden, dass er sowohl die Server-zum-Client-Verbindung als auch die Client-zum-Server-zurück-Verbindung bedeutet.

[0050] Der Server führt bzw. wartet verschiedene Informationen über die Übertragung und die Teilnehmer an der Übertragung. In der bevorzugten Ausführungsform werden diese Informationen vom Server in Form von Datenstrukturen oder Listen geführt bzw. gewartet. Der Server wartet und verwendet diese Informationen, um den Status der Dateiübertragung aufzuzeichnen und zu bestimmen.

[0051] Der Server führt bzw. wartet auch eine Frame-Datenstruktur, die alle selektiven Abweisungen für einzelne Frames von allen Clients angibt. Wenn mehrere Clients denselben Frame verpasst haben, würde die Frame-Datenstruktur nur angeben, dass der Frame verpasst wurde. Das heißt, die Frame-Datenstruktur wird nicht auf einer clientweisen Basis durch den Server gewartet. Es ist im Allgemeinen für den Server unerwünscht, eine detaillierte Liste von verpassten Frames auf einer Basis pro Client zu führen bzw. zu warten, da ein solches Schema eine ungeheure Menge an Speicher verwenden würde, insbesondere wenn eine große Anzahl (z. B. 1000 oder mehr) Clients an dem Multicast beteiligt sind. Es könnte beispielsweise sein, dass einer oder mehrere der Clients die Frames zwanzig und fünfundzwanzig von $Block_1$, den Frame eins von $Block_2$, bestimmte Frames von $Block_9$ usw. nicht empfangen oder fehlerhaft empfangen haben. Wenn der vom Server gewartete Frame-Status angibt, dass ein spezieller Frame eines speziellen Blocks erneut übertragen werden muss, ist es wahr, dass mindestens einer der Clients die erfolgreiche Vollendung dieses speziellen Blocks nicht

bestätigt hat. Nachdem der Server die ganze Datei einmal gesandt hat, würde der Server dann die Frame-Statusinformationen durchlaufen und nur die darin aufgelisteten Frames erneut senden. Dies würde Durchgang für Durchgang fortfahren, bis alle Clients "Fertigmeldungen" gesandt haben und die Frame-Statusliste leer ist (oder die maximale Anzahl von Runden oder die maximale Zeit erreicht wurde).

[0052] Es ist zu beachten, dass für irgendeinen gegebenen Durchgang, wenn irgendeine negative Bestätigung nicht zum Server zurück gelangt, der Client dieselben Abweisungs- und Neuübertragungs-Anforderungsmeldungen während des nächsten Durchgangs durch den Server zum Server zurücksendet. Dies bedeutet, dass, wenn ein bestimmter Client vom Server nicht gehört wird, dieser Client länger teilnehmen muss, aber sich dieser Client auf den Rest der empfangenen Clients nicht merklich auswirkt.

[0053] Ein weiteres Teil an Information bzw. Informationsstück, das am Server gespeichert wird, ist eine Statistik für die Multicast-Gruppe. Wenn eine Übertragung vollendet ist, werden Zusammenfassungsinformationen über die Übertragung bereitgestellt, die einen Operator beim Bestimmen von Systemleistungsproblemen und/oder der Leistungsprobleme eines speziellen Clients unterstützen können.

Mehrere Durchgänge durch die Datei:

[0054] Sobald die Datei vollständig einmal verarbeitet wurde (d. h. nach dem ersten Durchgang oder der ersten Runde) inkrementiert der erfindungsgemäße Übertragungsprozess einen Durchgangszähler und tastet dann die Frame-Statusliste im Server für den ersten Block ab, in dem ein Fehler bestand. Beim Finden des Blocks des ersten Fehlers sendet der Server die verpassten Pakete in diesem Block erneut. Negative Bestätigungen für diese verpassten Pakete werden, wie vorher beschrieben, von den Clients erzeugt, wenn sie einen Fehler in einem Block erfassen.

[0055] Dies ist mit dem ersten Durchgang konsistent. Alle selektiven negativen Abweisungsbestätigungen sind Angaben des Zustandes und sind daher für einen Durchgang nicht spezifisch, obwohl sie sich mit jedem Durchgang ändern können. In einer bevorzugten Ausführungsform liegen die mehreren selektiven negativen Abweisungsbestätigungen in Form von Bitmaps vor, wobei das ganze Wort einen Block darstellt und jedes Bit in diesem Wort einen anderen der Frames darstellt, die diesen Block bilden.

Übertragungsabbruch:

[0056] Wenn während der Übertragung ein Fehler angetroffen wird, der nicht berichtigt werden kann, oder wenn der Operator manuell abbricht, wird eine Übertragungsabbruchsequenz eingeleitet. Diese Sequenz hat die wiederholte Übertragung einer Abbruchmeldung für ein bestimmtes Intervall (z. B. für ein Intervall, das in einer Übertragungsdatei festgelegt ist) zur Folge. Die Empfänger bestätigen die Abbruchmeldung und können eine Handlung unternehmen, um beispielsweise entweder den Kontext für eine potentielle Wiederaufnahme (d. h. Neustart) der Übertragung zu speichern oder den Kontext zur Vorbereitung auf eine weitere Übertragung erneut zu initialisieren. Es gibt eine Einrichtung, die dem Anwender ermöglicht, einen Übertragungsabbruch einzuleiten. Ein Ursachencode kann gesetzt werden, um entweder zu unterbrechen oder zu initialisieren. Im ersteren Fall kann die Übertragung zu einer späteren Zeit wieder aufgenommen oder erneut gestartet werden, und im letzteren Fall werden die Clients aufgefordert, ihre Kontexte erneut zu initialisieren.

VOLLENDUNG

[0057] Der Server erfasst die Vollendung von einzelnen Clients durch Empfangen einer "Fertigmeldung" von einem Client.

[0058] Der Client weiß, dass er fertig ist, sobald er alle Blöcke der Datei hat, aber der Client muss weiterhin "Fertigmeldungen" senden, bis der Server die Vollendung bestätigt. Der Server bestätigt, dass ein Client "fertig" ist, indem er die Adresse dieses Clients in eine "Fertigliste" setzt und die Liste zu den Clients aussendet. Wenn ein Client seine Adresse in der "Fertigliste" aufgelistet sieht, weiß er, dass er die Übertragung vollendet hat. Der Client aktualisiert dann sein Übertragungsprotokoll, um anzugeben, dass die Übertragung erfolgreich vollendet wurde.

[0059] Eine Fähigkeit, eine Übertragung vom Server oder Client abzubrechen, ist enthalten. Ein Abbruchpaket versieht den Server und den Client mit der Fähigkeit, eine Übertragung vorzeitig abzubrechen. Wenn der Client einen Abbruch sendet, entfernt der Server den Client aus der Gruppe. Wenn der Server abbricht, kann die Übertragung ohne Senden der vollständigen Datei beim ersten Durchgang erneut gestartet werden.

Statusabfragen:

[0060] Wenn nach dem ersten Durchgang der Server weder ein FERTIG noch eine NAK vom Client empfangen hat, wird eine Abfrage gesandt, die an diejenigen Clients gerichtet ist, deren Status nicht bekannt ist. Die Antworten liegen in Form einer Standard-Antwortmeldung vor. Sie umfassen ein Bitmap, das die Fehler beschreibt, wenn Fehler zu melden sind.

Stau/Fluss-Steuerung:

[0061] Da große "Internets" Multicast-fähig werden, wird es üblicher, Multicast-Gruppen zu finden, die wünschen, dass Informationen verschiedene Übertragungsverbindungen zu den Mitgliedern dieser Gruppe haben. Diese verschiedenen Verbindungen können verschiedene Kapazitäten aufweisen, die erheblich voneinander abweichen können. Ein Mitglied der Gruppe kann beispielsweise eine Verbindungskapazität von über 1 Mbps aufweisen, während ein anderes nur 56 Kbps aufweisen kann. Im Allgemeinen ist die Kenntnis dieser Verbindungskapazitäten durch den Sender (z. B. den Server) der Übertragung nicht bekannt. Folglich ist es erwünscht, die Verbindungskapazitäten im Fluge bestimmen zu können und einen Flusssteuermechanismus bereitzustellen, um eine Überlastung/einen Stau des Netzwerks zu verhindern, während gleichzeitig die Effizienz des Datenübertragungsprotokolls nicht gehemmt wird.

[0062] Das hierin beschriebene Datenübertragungsprotokoll umfasst das Konzept von Blöcken, von denen jeder Hunderte oder Tausende von Frames enthalten kann. Clients (Rezipienten) sind verpflichtet, eine mehrfache selektive Abweisungs-NAK an Blockgrenzen zu senden, wenn irgendwelche Frames fehlen oder in diesem Block fehlerhaft sind. Für Flussteuerzwecke ist es erwünscht, Kenntnis von verpassten/fehlerhaften (d. h. fallen gelassenen) Frames sobald wie praktisch zu erlangen, so dass Flussteuerentscheidungen getroffen werden können. Sich ändernde oder variable Blockgrößen ist eine Weise zum Durchführen dessen mit dem Datenübertragungsprotokoll der Erfindung und dies beinhaltet das Starten mit einem relativ kleinen Block und das Vergrößern der Blockgröße während der Dateiübertragung, um die aktuelle Skalierbarkeit beizubehalten, indem Client-Bestätigungen verringert werden. Eine weitere, bevorzugte Weise zum Durchführen dessen mit dem Datenübertragungsprotokoll der Erfindung besteht darin, die Blockgrößen alle gleich zu halten (homogene Blockgrößen), aber den Server Statusabfragen aussenden zu lassen, bevor eine Blockgrenze auftritt, so dass Clients vor der Blockgrenze mit NAKs antworten. Dieses letztere Verfahren ist das flexibelste, da NAKs jederzeit abgefragt werden können, im Gegensatz zu nur an Blockgrenzen, was beim ersteren Verfahren der Fall ist. Mit einem dieser zwei Verfahren werden NAKs früh bei der Übertragung abgefragt.

[0063] In dem Verfahren mit "variabler Blockgröße" (das erstgenannte Verfahren im vorangehenden Absatz) kann der erste Block relativ klein, z. B. 100 Frames, sein. Nachfolgende Blöcke nehmen jedes Mal um einen Faktor von Zwei zu. Die Blöckgrößen werden zunehmend verdoppelt, bis die maximale Blockgröße erreicht ist oder die Datei ihr Ende erreicht.

[0064] In dem "Statusabfrage"-Verfahren (das vorher erwähnte zweite Verfahren) fragt der Server NAK-Antworten an Punkten ab, wo er es wünscht, und diese Punkte liegen nicht an Blockgrenzen. Mit dieser bevorzugten Ausführungsform für die Stau- oder Flussteuerung werden Statusabfragen in zunehmend längeren Intervallen gesandt.

[0065] Mit beiden Verfahren wird die Übertragungsrate oder Transferrate festgelegt, wie hierin beschrieben. Anstatt einer festen Übertragungsrate stellt jedoch die einstellbare Rate eine obere Grenze für die Übertragungsrate dar. Nach dem ersten Block (mit dem Verfahren mit variabler Blockgröße), oder wenn eine Statusabfrage empfangen wird (mit dem Statusabfrageverfahren), werden NAKs durch Clients zum Server gesandt, die Frames fallen gelassen haben, was eine Stauangabe durch diese Clients an den Server darstellt.

[0066] Wenn NAKs vorliegen, kann die Tatsache, dass sie direkt mit der momentanen Kapazität der speziellen Verbindung in Beziehung stehen, verwendet werden, um die Verbindungskapazität für alle Verbindungen, die einen Stau zeigen, auf der Basis der folgenden Gleichung zu bestimmen: ((# gesandte Frames – # Frames mit NAK)/# gesandte Frames) Übertragungsrate = Verbindungskapazität. Im heterogenen Multicast-Netzwerk von **Fig. 6** liegen die Verbindungsgeschwindigkeiten im Bereich von 64 Kbps bis 1024 Kbps, eine große Differenz in der Verbindungskapazität. Wenn kein weiterer Verkehr angenommen wird und wenn die Übertragungsrate für 150 Kbps eingestellt wäre, würde die Block-NAK für den Block 1 vom Client A (in dem Verfahren mit variabler Blockgröße) etwa 58 fallen gelassene Frames für den ersten Block angeben. Unter Verwendung der obigen Gleichung wird die momentane Verbindungsgeschwindigkeit als 63 Kbps berechnet. Die Block-NAK vom Client B für den Block 1 (in dem Verfahren mit variabler Blockgröße) würde etwa 15 fallen gelassene Fra-

mes für den ersten Block angeben. Wieder unter Verwendung der Gleichung wird die momentane Verbindungsgeschwindigkeit für die Verbindung zu B als 127,5 Kbps berechnet. Wenn weiterer Verkehr vorhanden ist, wäre die Anzahl von fallen gelassenen Frames höher, was zu einer kleineren berechneten Verbindungsgeschwindigkeit führt.

[0067] Ein Gruppenschwellen-Parameter kann vom Anwender eingestellt werden. Die Gruppenschwelle ist die Grenze, in Prozent von fallen gelassenen Frames ausgedrückt, durch einen speziellen Client, der für eine fortgesetzte Teilnahme an der Multicast-Gruppe zugelassen ist. Wenn die Gruppenschwelle auf 25 % gesetzt wird, bedeutet dies, dass irgendwelche Clients in der Gruppe, die einen höheren Frame-Ausfallprozentsatz als 25 % haben, eine Handlung unternehmen müssen, so dass der Rest der Gruppe nicht nachteilig beeinflusst wird. In dem Beispiel von Fig. 6 müsste der Client A mit 58 % von fallen gelassenen Frames eine Handlung unternehmen. Clients haben genügend Informationen, um diese Entscheidung zu treffen, da die Übertragungsraten- und Gruppenschwellenparameter zu Clients in der Ankündigungsmeldung übertragen werden. Clients, die erfassen, dass ihre Frame-Ausfallrate die Schwelle übersteigt, können eine der folgenden Handlungen unternehmen:

1. Verlasen der Gruppe und Anfordern vom Server, dass sie in eine Gruppe mit niedrigerer Geschwindigkeit gesetzt werden, wobei die Gruppengeschwindigkeit auf der Basis der an diesem Client durchgeführten Messung festgelegt wird;
2. Verlassen der Gruppe ohne Anfordern einer weiteren Lieferung, was bedeutet, dass dieser Client diese Übertragung verpasst; und
3. Unterdrücken von NAKs, bis eine Statusabfragemeldung vom Server empfangen wird, was ermöglicht, dass der Rest der Gruppe beendet, ohne durch übermäßige Neuübertragungen von einem Client mit hohem Frame-Verlust aufgehalten zu werden (die Übertragungsrate für Neuübertragungen zu diesem Satz von Clients könnte niedriger sein, um ihre niedrigeren Kapazitäten widerzuspiegeln).

[0068] In dem Beispiel von Fig. 6 kommt der nächsthöchste Prozentsatz von Frame-Ausfällen vom Client B mit 15 %, was unter der Gruppenschwelle liegt. Diese Zahl stellt einen Faktor dar, auf den sich die ganze Gruppe ohne übermäßige Leistungsverschlechterung einstellen kann. Die Übertragungsrate des Servers für die Gruppe wird dann um 15 % oder einen höheren oder geringfügig höheren Prozentsatz gesenkt, um dem Client B gerecht zu werden.

[0069] Die Zeitsteuerung des Verfahrens mit variabler Blockgröße ist in Fig. 8 angegeben. Sobald Informationen an einem Client angeben, dass seine Frame-Ausfälle die Gruppenschwelle übersteigen, muss dieser Client eine der drei vorstehend aufgelisteten alternativen Handlungen unternehmen, so dass die Gruppenübertragung nicht nachteilig beeinflusst wird. Die Einstellung der Übertragungsrate der Gruppe wird durchgeführt, nachdem der zweite Block gesandt wurde, wobei mit dem Beginn von Block 3 begonnen wird. Übertragungsratenänderungen werden an Blockgrenzen implementiert, um genaue Daten auf einer Blockbasis von den Block-NAKs zu liefern. Die Dateiübertragung geht dann mit der Übertragung von Block 3 weiter, der als zweimal so groß wie der Block 2 eingestellt ist, ebenso wie der Block 2 zweimal so groß ist wie der Block 1. Diesem folgt Block 4, der zweimal so groß ist wie der Block 3, und so weiter, bis die maximale Blockgröße erreicht ist oder die Datei ihr Ende erreicht, was auch immer zuerst geschieht. Wenn NAKs von der Gruppe nach dem Block 3 jedoch angeben, dass der schlechteste Client einen Ratenschwellenparameter (konfigurierbar) übersteigt, dann wird die Rate für die Übertragung des Blocks 5 weiter eingestellt. Die Ratenschwelle ist der minimale Frame-Ausfallprozentsatz, für den Übertragungsrateneinstellungen für die Gruppe durchgeführt werden. Ein maximaler Frame-Ausfallprozentsatz von 1 % von den Clients würde beispielsweise keine Einstellung garantieren, so dass die Ratenschwelle typischerweise auf eine Zahl oberhalb 1 % gesetzt werden würde.

[0070] In dem Statusabfrageverfahren weisen die Blöcke eine gleichmäßige Größe auf und Statusabfragen werden vom Server zu Abfrage-NAKs gesandt, bevor Blockgrenzen erreicht sind. Mit Bezug auf Fig. 9 ist ein äquivalentes Szenario zu dem gerade beschriebenen für das Verfahren mit Blöcken mit variierender Größe dargestellt, außer dass nun (in Fig. 9) die Blockgrößen homogen sind. In einem Beispiel wird die erste Statusabfrage nach 100 Frames der Übertragung gesandt, die zweite, nachdem 200 weitere Frames gesandt sind, usw. Client-NAKs werden zu exakt denselben Zeiten zum Server zurückgesandt wie beim Verfahren mit variabler Blockgröße. Es besteht jedoch die hinzugefügte Flexibilität bei dem Statusabfrageverfahren, dass Statusabfragen vielmehr zu einer beliebigen gewünschten Zeit gesandt werden, als auf eine Blockgrenze warten zu müssen, um NAKs zu empfangen, wie es bei dem Verfahren mit variabler Blockgröße der Fall ist.

[0071] Bei entweder dem Verfahren mit variabler Blockgröße oder dem Statusabfrageverfahren ist es im Allgemeinen nicht erwünscht, einfach Gruppenmitglieder zu löschen und sie hängen zu lassen. Gelöschte Gruppenmitglieder können in einer anderen Gruppe gesammelt werden, die mit einer niedrigeren Übertragungsrate

arbeitet. Diese niedrigere Übertragungsrate kann durch die Berechnung an den Verbindungskapazitäten bestimmt werden, die durch Clients durchgeführt wird, die die Gruppe verlassen. Diese Gruppe kann dann mit einer entsprechenden Übertragungsrate eingerichtet werden und eine neue Übertragung kann eingeleitet werden.

[0072] Sowohl das Verfahren mit variabler Blockgröße als auch mit Statusabfrage des Flusssteuerprozesses können automatisch durchgeführt werden.

Multicast:

[0073] Multicast kann in zwei Formen vorliegen: Anwendungsschicht-Multicast (AL-Multicast), wobei das Netzwerk immer noch Daten zur gesamten Broadcast-Gruppe liefert, und Multicast-IP, wobei das Netzwerk Verkehr auf der Basis von Multicast-Routern leitet und die Internet-Spezifikation RFC 1112 in den Clients implementiert wird.

[0074] In beiden Fällen werden Multicast-Gruppen auf Betreiben des Servers eingerichtet. Der Server sendet Benachrichtigungen auf einer Unicast-Basis zu Clients, um sie sie über die Mitgliedschaft in einer speziellen Multicast-Gruppe zu informieren. Diese Multicast-Gruppen können schnell eingerichtet und aufgelöst werden, was eine dynamische Konfiguration von Multicast-Gruppen ermöglicht. Eine Multicast-Gruppe könnte beispielsweise so eingerichtet werden, dass sie nur für die Übertragung einer speziellen Datei existiert, nach welcher Zeit die Gruppe aufgelöst werden würde.

[0075] Mit AL-Multicast liefert das Netzwerk immer noch Verkehr auf einer Broadcast-Basis, aber Clients, die sich nicht in der Gruppe befinden, verwerfen die Daten, die nicht für sie bestimmt sind. Wenn die Gruppe eingerichtet wird, können auch Sicherheitsschlüssel verteilt werden, so dass Clients außerhalb der Gruppe die Daten nicht lesen können, selbst wenn es passieren würde, dass die Daten an diesem Knoten nicht verworfen werden würden (es ist zu beachten, dass dies auch bei Multicast-IP eingesetzt werden könnte). Bei AL-Multicast bleibt die IP-Adresse auch eine globale oder Netzwerkbasis-Broadcast-Adresse. Wie beim Broadcast wird diese Adresse auf eine Broadcast-Adresse im Verbindungsschichtprotokoll abgebildet, z. B. eine Broadcast-SMDS-Adresse. Ein Multicast-Kopf wird für die Gruppe ausgewählt und wird zum Gruppendifferentiator.

[0076] Bei Multicast-IP ist das Netzwerk ein Router-Netzwerk, wobei die Router Multicast-IP-Adressen der Klasse D und Multicast-Leitung unterstützen. Die Clients unterstützen RFC 1112, "Hauptrechnererweiterungen für IP-Multicasting". RFC 1112 sieht eine Hauptrechnerbenachrichtigung über ihre Anwesenheit für den nächsten Multicast-Router für den Zweck der Aktualisierung der Router-Tabellen vor.

[0077] Mit Rückbezug auf **Fig. 2**, die im Allgemeinen ein beliebiges Broadcast- oder Multicast-IP-Netzwerk auf Router-Basis darstellen kann, besteht ein Zweck der Erfindung darin, die gleichzeitige Übertragung von kleinen oder großen Dateien (z. B. Dateien bis zu einer Größe von 2 Gigabytes oder mehr) durch einen Server **20** zu bis zu 5000 oder mehr Empfangsknoten **22** über eine Verbindung **24** eines weiträumigen Netzwerks (WAN) zu ermöglichen. Die Erfindung kann auch über lokale Netzwerke und andere Arten von Kommunikationsverbindungen arbeiten, wie vorher beschrieben. Das Übertragungsmedium **24** kann eine beliebige Art sein, das das TCP/IP-Protokollprofil in der bevorzugten Ausführungsform unterstützt. Andere Protokollprofile könnten auch als Kommunikationsumgebung für die Erfindung dienen.

[0078] Multicast kann in zwei Weisen unterstützt werden AL-Multicast und Multicast-IP, wie vorher erwähnt.

[0079] Zu den Clients zu übertragende Dateien können über ein Band (z. B. das Bandlaufwerk **66** von **Fig. 5**) oder, wenn die Dateien klein genug sind, durch eine Diskette (z. B. das Diskettenlaufwerk von **Fig. 5**) auf den Server **20** geladen werden. Zu übertragende Dateien können auch über FTP (File Transfer Protocol) oder irgendeinen anderen Unicast-Übertragungsmechasnismus von der Quelle der Datei über ein LAN oder anderes Netzwerk beispielsweise auf den Server **20** geladen werden. Die Dateien können im Allgemeinen in einem beliebigen Format vorliegen. Die Datei wird dann vom Band oder von der Diskette in ein Dateisystem des Übertragungsservers **20** eingelesen. Es ist zu beachten, dass der Server **20** ausreichend Platz verfügbar haben muss, um eine unkomprimierte Kopie der Datei einzulesen. Für beide Dienste kann die Datei auch verschlüsselt werden, so dass unberechtigte Empfänger die Datei nicht empfangen und verwenden können. Jede Übertragungsdatei ist vorzugsweise eindeutig identifiziert. Es besteht vorzugsweise eine Angabe hinsichtlich ihres Inhalts und ihrer Erzeugungszeit. Die Eingangsdateien in den Prozess können eine Größe von über 2 Gigabytes aufweisen und das System kann auch Dateien bearbeiten, die viel größer als 2 Gigabytes sind.

[0080] Die Datei kann dann auf dem Server **20** gespeichert und zur Übertragung vorbereitet werden. Daten von vorherigen Übertragungen müssen auf dem Server **20** für einen gewissen Zeitraum leicht erhältlich sein, falls sie erneut übertragen werden müssen. Ein Mechanismus zum Zugreifen auf die Daten ist derart vorgesehen, dass die Daten leicht für die erneute Übertragung in eine Warteschlange eingereiht werden können.

[0081] Wegen der Effizienz wird die Datei in Blöcken übertragen. Die Größe eines Blocks wird vom größten Paket abgeleitet (oder die Blockgröße kann vom Anwender ausgewählt werden), das über den Kommunikationsweg **24** übertragen werden kann. Ihre Ableitung basiert auf der Tatsache, dass die Clients dem Server angeben müssen, welche der Pakete in einem Block sie nicht empfangen haben. Eine Art und Weise und im Allgemeinen die einfachste Weise, dies durchzuführen, besteht darin, ein Bitmap zu senden, das durch ein in der Position eingestelltes Bit angibt, welche Pakete nicht empfangen wurden. Die Größe des Blocks ist daher ungefähr die Anzahl von Paketen, die in einem Bitmap bestätigt werden kann, das selbst in einem Paket enthalten sein kann. Wenn beispielsweise die Paketgröße 256 Bytes wäre, dann wären die meisten Bits, die ein Paket enthalten könnte, 256 (Bytes/Paket)·8 (Bits/Byte) = 2048 (Bits/Paket), was bedeutet, dass die größte zulässige Blockgröße ein Block mit 2048 Paketen wäre.

[0082] Obwohl die empfangenden Knoten **22** mit einem Ethernet-LAN mit 10 Mbps gekoppelt sein können, besitzen WAN-Verbindungen häufig viel niedrigere Geschwindigkeiten als diese. Folglich ist eine explizite Übertragungsdatenrate einstellbar/konfigurierbar.

[0083] Die empfangenden Knoten können jeweils Ressourcenprobleme entweder vor oder während einer Übertragung erfahren. Den empfangenden Knoten wird ermöglicht, ihre Ressourcen vor einer Übertragung abzufragen und festzustellen, ob sie die Einrichtungen zum Empfangen der Daten haben. Wenn nicht, dann sollten sie entweder Raum, der für die Übertragung zweckgebunden ist, erneut initialisieren oder sollten angeben, dass sie nicht an der Übertragung teilnehmen können, und eine Korrekturmaßnahme kann durch verschiedene Kanäle unternommen werden. Eine Einrichtung könnte bereitgestellt werden, wobei der Server die Verfügbarkeit von Plattenplatz entfernt erzwingen kann, um zu ermöglichen, dass die Übertragung der Datei stattfindet.

[0084] Die Empfänger **22** müssen sich auch dessen bewusst sein, worauf sie hören. Wenn ein Datagramm auf einem zweckgebundenen Kanal empfangen wird, muss der Knoten **22** feststellen, ob er adressiert wird. Ein Problem kann entstehen, wenn diese Anwendung von mehr als einem Übertragungsserver **20** verwendet wird. Es muss eine Weise zum Garantieren, dass ein empfangender Knoten **22** an exakt einer Übertragung zu einer gegebenen Zeit teilnimmt, bestehen. Durch Reservieren eines UDP-Ports für einen Server **20** und auch Verbinden eines Verschlüsselungsschlüssels mit diesem Server wird sichergestellt, dass ein empfangender Knoten, der einen vermischten Modusabgriff am Netzwerk **24** verwendet, nicht die Fähigkeit hat, die übertragenen Daten interpretieren zu können.

[0085] Einige Bezugsinformationen werden auf dem Übertragungsserver **20** gewartet. Es befindet sich vorzugsweise eine Liste aller potentiellen empfangenden Knoten im Netzwerk. Genügend Bezugsinformationen stehen vorzugsweise zur Verfügung, um zu ermöglichen, dass der Informationsanbieter die Clients im Fall von Dienstausfällen, Problemen usw. managt. Vorzugsweise ist eine Übertragungsdatenbank vorhanden, in der eine verschlüsselte komprimierte Datei zur Übertragung bereit gehalten wird. Die Übertragungsdatenbank enthält die vorbereiteten Daten zusammen mit Beschreibungsinformationen von beispielsweise bis zu 70 Bytes, die den Inhalt der Dateien identifizieren.

[0086] Jede Übertragung hat vorzugsweise einen Vollendungsstatus-Indikatordatensatz und ein Protokoll aller Fehler, die während der Übertragung angetroffen werden. Vorzugsweise ist auch eine Ereignisdatei mit einer Liste aller Knoten, für die die Übertragung fehlgeschlagen ist, wer aufgerufen werden soll, und warum sie fehlgeschlagen ist, vorhanden.

[0087] Zu irgendeinem Zeitpunkt während der Übertragung kann ein Operator den Status der Übertragung abfragen, während sie auf dem Server **20** und jeden der empfangenden Knoten **22** angewendet wird. Alarme werden erzeugt, wenn Probleme bei der Kommunikation mit bestimmten Clients oder andere Probleme bestehen. Wenn irgendein Eingriff angegeben wird, wird dem Operator ermöglicht, die Korrekturhandlung einzuleiten.

[0088] Für eine weitergehende Wartung und ein weitergehendes Management des Dienstes wird dem Operator ermöglicht, die Liste von Empfängern, Übertragungsgruppen, Übertragungsdatei-Deskriptoren, Übertragungsparametern und Übertragungsdatenbanken zu warten. Ein Hintergrundprozess wartet die Umgebung und sowohl alte Daten und löscht sie auch gemäß Organisationsparametern, falls durch einen alarmierten

Operator ermöglicht.

[0089] Die erfindungsgemäße Datenübertragung wurde vorstehend beschrieben. Weitere Aspekte der Erfindung werden nachstehend beschrieben. Diese weiteren Aspekte umfassen: EINSTELLBARE ÜBERTRAGUNGSRATE; MULTICAST-GRUPPEN; MULTICAST-PING; MULTICAST-NETZWERKSONDE; GESCHWINDIGKEITSGRUPPEN; und SAMMLUNG NEGATIVER BESTÄTIGUNGEN.

EINSTELLBARE ÜBERTRAGUNGSRATE

[0090] Wie vorher erwähnt, ist es möglich, die Datenübertragungsrate einzustellen. Das vorher gegebene Beispiel stellte dar, wann eine einstellbare Rate nützlich ist. In diesem Beispiel sind die empfangenden Knoten **22** mit einem Ethernet-LAN mit einer verfügbaren Bandbreite von 10 Mbps gekoppelt und die WAN-Verbindungen, die das LAN mit anderen Netzwerken verbinden, besitzen viel niedrigere Geschwindigkeiten als 10 Mbps. In einem solchen Fall würde die Datenübertragungsrate zur Anpassung beispielsweise auf die Geschwindigkeit der langsamsten WAN-Verbindung gesetzt werden.

[0091] Für irgendeine gegebene Dateiübertragungssitzung kann die Datenübertragungsrate im Allgemeinen im Voraus festgelegt werden. Insbesondere ist die maximale Bitrate, mit der Daten während der Sitzung übertragen werden, einstellbar. In einer bevorzugten Ausführungsform wird sie durch Einstellen eines Parameters auf einen ganzzahligen Wert, der die Bitrate in Kilobits pro Sekunde (Kbps) darstellt, eingestellt. Wen dieser Ratenparameter beispielsweise einen Wert von 56 aufweist, entspricht er einer maximalen Bitrate von 56 Kbps. Der Ratenparameter kann auf einen beliebigen Wert gesetzt werden, der der verfügbaren Bandbreite der Verbindung entspricht, die die Quelle mit dem Zielort verbindet, oder auf einen Wert, der eine Rate darstellt, die geringer ist als die verfügbare Bandbreite. Das heißt, wenn die verfügbare Bandbreite 1 Mbps ist, kann der Ratenparameter auf irgendeinen Wert zwischen Null und eintausend gesetzt werden, wobei 1000 Kbps gleich 1 Mbps ist. Diese Fähigkeit, die Übertragungsrate explizit einzustellen, ermöglicht, dass lange (zeitlich) Dateiübertragungen gleichzeitig mit anderen Anwendungen im Netzwerk existieren, ohne die ganze oder im Wesentlichen ganze Bandbreite des Netzwerks zu belegen.

MULTICAST-GRUPPEN

[0092] "Multicast" ist vorstehend als der Fall definiert, in dem der Serverknoten **20** Daten (z. B. eine Datei) zu einer Untermenge aller Client-Knoten **22** sendet, die mit dem Netzwerk **24** verbunden sind. Vorstehend ist auch offenbart, dass die Multicast-Übertragung in zwei Formen vorliegen kann: "Anwendungsschicht-Multicast (AL-Multicast)" und "Multicast-IP". AL-Multicast wird verwendet, wenn das Netzwerk die Internet-Spezifikation RFC 1112 nicht unterstützt, aber Broadcast unterstützt. Wenn Multicast-IP durch das Netzwerk gemäß RFC 1112 und Multicast-IP-Routing unterstützt wird, ist es gegenüber AL-Multicast empfohlen. Multicast-IP wird verwendet, wenn Mitglieder der Gruppe Multicast unterstützen müssen und Router im Router-Netzwerk auch eine gewisse Art eines Multicast-Leitungsprotokolls (z. B. DVMRP, MOSPF oder PIM) unterstützen müssen. Im Gegensatz zu AL-Multicast ist Multicast-IP ein echtes Multicast-Protokoll, bei dem nur Mitglieder der Multicast-Gruppe die übertragenen Daten empfangen.

[0093] Für jede Dateiübertragung kann eine Multicast-Gruppe während des ANKÜNDIGUNGS/REGISTRIERUNGS-Aspekts der Datenübertragung definiert werden, wie vorstehend beschrieben. Wie angegeben, wartet der Server verschiedene Informationen über die Dateiübertragung und die Teilnehmer oder die Gruppe, die an der Übertragung beteiligt ist/sind. In der bevorzugten Ausführungsform werden diese Informationen durch den Server in Form von Datenstrukturen oder Listen gewartet. Der Server wartet und verwendet diese Informationen, um den Status der Dateiübertragung während der DATENÜBERTRAGUNGS-Stufe aufzuzeichnen und zu bestimmen. Die Client-Statusstruktur umfasst eine Liste des Status der Teilnehmer der Multicast-Gruppe auf der Basis von Daten von den Ankündigungsregistrierungen, die vom Server empfangen werden.

[0094] Das Multicast-Gruppenmanagement ist der Prozess des Zuweisens von Clients zu Multicast-Gruppen. Die Aufgabe der Organisation und der Behandlung der Liste von Clients in jeder Gruppe ist die Verantwortung des Anwendungsprogramms, das die Dateiübertragung im ersten Fall einleitet. Das Anwendungsprogramm sieht im Allgemeinen leicht zu verwendende Merkmale vor, wie z. B. Zuweisen eines Namens zu einer Client-IP-Adresse, Zuweisen eines Namens zu einer Gruppe usw. Das Gruppenmanagement ist nur an der sendenden Station, z. B. am Server, erforderlich. Eine Multicast-Gruppe wird festgelegt, wenn eine sendende Station eine Datei übertragen will. Die Gruppe wird durch eine Liste von Client-IP-Adressen identifiziert, eine Adresse für jeden Client in der Multicast-Gruppe.

[0095] Es gibt zwei Optionen für Multicast-Gruppen dynamisch und statisch. Für dynamische Multicast-Gruppen löst sich die Gruppe auf, wenn die Übertragung vollendet ist. Dynamische Multicast-Gruppen werden mit den ANKÜNDIGUNGS-Meldungen unter Verwendung von Multicast-Gruppen-Adressen der Klasse D gebildet. Im Gegensatz zu dynamischen Multicast-Gruppen bleiben bei statischen Multicast-Gruppen alle Mitglieder der Gruppe Mitglieder der Gruppe, wenn die Übertragung vollendet ist. Statische Multicast-Gruppen werden vom Server auf einer Unicast-Basis und/oder unter Verwendung einer gemeinsamen Adresse der Klasse D gebildet, um Konfigurationen einzurichten.

MULTICAST-PING

[0096] Das "Ping"-Dienstprogramm in TCP/IP ist beim Bestimmen der Konnektivität zwischen zwei Punkten in einem TCP/IP-Netzwerk (d. h. beim Bestimmen, ob zwei Punkte tatsächlich verbunden sind) sehr nützlich. Bei TCP/IP wird ein Ping-Paket zum gewünschten Endpunkt gesandt, der die Adressen umkehrt und sie zum Sender zurücksendet. Die Umlaufzeitverzögerung wird auch gemessen und dies ist eine Messung der Zeit, die es dauert, bis das Ping-Paket vom Sender zum gewünschten Endpunkt und dann zum Sender zurückläuft.

[0097] Es ist auch erwünscht, ein Multicast-Ping-Dienstprogramm bereitzustellen, wobei alle Mitglieder einer Multicast-Gruppe auf das Ping-Paket oder die Ping-Anfrage antworten. Clients oder Hauptrechner, die Multicast-IP (RFC 1112) unterstützen, antworten auf eine Ping-Anfrage mit einer IP-Adresse der Klasse D als Zieladresse. In bekannten Multicast-Implementierungen zeigt jedoch der Sender der Ping-Anfrage nur die erste Antwort, die er auf seine Ping-Anfrage empfängt, an. Das heißt, bekannte Multicast-Ping-Verfahren führen keine Netzwerk-Konnektivitätsmessung durch.

[0098] Ein "Multicast-Ping"-Merkmal der Erfindung zeigt alle Multicast-Antworten auf eine Ping-Anfrage an, wodurch die Netzwerkkonnektivitäts-Informationen von der Quelle zu den Gruppenrezipienten und die Umlaufzeitverzögerungsinformationen für jeden Multicast-Gruppenrezipienten geliefert werden. In einer bevorzugten Ausführungsform verwendet dieses Merkmal die Standard-Ping-ICMP-Meldungen.

[0099] Als erfindungsgemäße Verbesserung ist es auch möglich, die vorher beschriebene Ankündigungs/Registrierungs-Einrichtung als weitere Form des "Multicast-Ping"-Merkmals zu verwenden. Mit dieser Verbesserung bestimmen Ankündigungs/Registrierungs-Ping-Meldungen die Konnektivität mit der Anwendungsschicht der Gruppenrezipienten und wieder zum Sender und die Umlaufzeitverzögerungsinformationen für jeden Gruppenrezipienten.

[0100] Das "Multicast-Ping"-Merkmal ermöglicht folglich, dass eine Netzwerkkonnektivität und Umlaufverzögerungen durch den Sender für Mitglieder einer Multicast-Gruppe bestimmt werden.

MULTICAST-NETZWERKSONDE

[0101] Multicast- (das Senden von einem zu vielen, aber nicht zu allen) Datennetzwerke beginnen gerade jetzt implementiert zu werden. Multicast-IP insbesondere ist in Router-Netzwerken neu und kann den Mechanismus zum Erzeugen von Multicast-Gruppen über Netzwerken aller Arten bereitstellen, z. B. Frame-Relay, SMDS, LANs, Satelliten, drahtlos. Das Internet hat auch das "Mbone" (Multicast-Gerüst), ein Teil des Internets, der Multicast-IP unterstützt.

[0102] Das Mbone wurde Anfang 1992 gestartet und ist gewachsen, so dass am Beginn von 1995 über 1500 Unternetze des Internets Multicast-fähig waren. Bis zu diesem Punkt wurde das Mbone als experimentelles Netzwerk durch Internet-Forscher verwendet, die Videokonferenz auf PC- und Arbeitsplatzrechnerbasis und Whiteboard-Multicast-Anwendungen sowie Internet-"Radio" und andere experimentelle Anwendungen getestet haben. Multicast-IP-Leitung auf dem Mbone wurde anfänglich in Arbeitsplatzrechnern unter Verwendung des Multicast-Leitungsprotokolls DVMRP implementiert; bei Teilen des Mbone wurden jedoch die Router aufgerüstet, so dass sie Multicast-fähig waren. Es wird erwartet, dass innerhalb 5 bis 6 Jahren das Internet unter Verwendung der Router im Internet vollständig Multicast-fähig ist.

[0103] Wenn mehr des Internets Multicast-fähig wird, wird es vielmehr für Haupt-Multicast-Anwendungen als als experimentelles Forschungswerkzeug verwendet. Wenn dies geschieht, sind Werkzeuge erforderlich, um die Verwendung zu erleichtern.

[0104] Ein großer Unterschied zwischen dem Internet und einem privaten Netzwerk besteht darin, dass das Internet ein sehr heterogenes Netzwerk ist. Es ist ein Netzwerk von Netzwerken und es bestehen große Un-

terschiede in den verschiedenen Teilen des Netzwerks, die von verschiedenen Organisationen betrieben werden. Im Gegensatz dazu werden viele private Netzwerke als relativ homogen mit viel Steuerung durch den Operator des privaten Netzwerks hinsichtlich der Architektur des Netzwerks eingerichtet.

[0105] Da die Endpunkte im Multicast-Netzwerk wahrscheinlich mit verschiedenen Raten mit verschiedenen Netzwerken verbunden sind und ein Stau im Netzwerk in verschiedenen Teilen des Netzwerks unterschiedlich ist, ist es erwünscht, die Kenntnis der Kapazität der beigefügten Verbindungen in der Multicast-Gruppe erlangen zu können und die Leistung bei dieser Kapazität zu testen. Ein "Multicast-Netzwerksonden"-Merkmal der Erfindung ist so ausgelegt, dass es das Mbone oder ein anderes großes heterogenes Multicast-Netzwerk von der Verkehrsquelle aus prüfen und die Kapazität der einzelnen Verbindungen schnell von dieser Verkehrsquelle aus messen kann.

[0106] Mit Bezug auf Fig. 6 hat ein heterogenes Multicast-Netzwerk (z. B. der Mbone-Abschnitt des Internets) eine Multicast-Gruppe mit fünf Mitgliedern A bis E, wobei jedes Mitglied der Gruppe durch eine Verbindung mit anderer Kapazität, d. h. eine Verbindung mit anderer Rate, verbunden ist. Das Mitglied A der Gruppe ist mit dem Netzwerk mit einer Verbindung mit 64 Kbps (Kilobits pro Sekunde) verbunden, B mit einer Verbindung mit 128 Kbps, C mit einer Verbindung mit 256 Kbps, D mit einer Verbindung mit 512 Kbps und E mit einer Verbindung mit 1024 Kbps. Die Art dieser Verbindungen ist dem Server (d. h. der Verkehrsquelle) unbekannt, da Verbindungen mit dem Internet mit vielen Verbindungen mit unterschiedlicher Geschwindigkeit vorhanden sein können.

[0107] Es ist erwünscht, dass die Verkehrsquelle die Eigenschaften der Verbindungen zu den Zielorten kennt, so dass sie optimal bestimmen kann, wie die Multicast-Übertragung von Informationen zu den Zielorten durchzuführen ist. Wenn die Anwendung eine Videokonferenz ist, kann festgestellt werden, dass die Qualität für A mit 64 Kbps unannehmbar sein kann, aber der Rest mit 128 Kbps teilnehmen könne. Wenn die Datenübertragung die Anwendung ist, könnten ebenso die Gruppen D und E eine Gruppe bilden, die mit einer Übertragungsrate von 512 Kbps arbeitet, während die Gruppen A, B und C mit 64 Kbps arbeiten könnten, ohne die Netzwerkkapazität zu überschreiten.

[0108] Gemäß der Erfindung ist der Mechanismus zum Prüfen des Netzwerks, um die Kapazität der entfernten Verbindung zu bestimmen, das hierin beschriebene System und Protokoll. Nach der Ankündigung/Registrierung wird eine Multicast-Gruppe von Mitgliedern A bis E als Mittel verwendet, um die Konnektivität gemäß beispielsweise dem "Multicast-Ping"-Merkmal der Erfindung, das im vorangehenden Abschnitt beschrieben ist, zu bestimmen (d. h. zu bestimmen, welche Mitglieder tatsächlich mit dem Server verbunden sind), eine Testreihe von kleinen Dateien wird der Reihe nach mit verschiedenen Geschwindigkeiten zu den Gruppenmitgliedern gesandt. Eine Testdatei mit 400 Frames kann beispielsweise zuerst mit 64 Kbps, dann 128 Kbps, dann 256 Kbps, dann 512 Kbps und schließlich mit 1024 Kbps gesandt werden. Negative Bestätigungen der Clients werden empfangen und am Server gespeichert, wie in nachstehender Tabelle 1 gezeigt, wobei kein weiterer Verkehr auf den Verbindungen angenommen wird.

Tabelle 1 – Testergebnisse mit einer Testdatei mit 400 Frames

Sendegeschwindigkeit	# Naks für A	# Naks für B	# Naks für C	# Naks für D	# Naks für E
64 Kbps	0	0	0	0	0
128 Kbps	200	0	0	0	0
256 Kbps	300	200	0	0	0
512 Kbps	350	300	200	0	0
1024 Kbps	375	350	300	200	0

[0109] Mit Bezug auf Tabelle 1 führt der erste Durchlauf mit einer Geschwindigkeit von 64 Kbps zu keinen negativen Bestätigungen (d. h. NAKs oder Naks) für irgendeines der Gruppenmitglieder, da alle Verbindungen 64 Kpbs oder mehr unterstützen.

[0110] Der zweite Durchlauf geschieht mit 128 Kbps, zweimal jene der ersten. In diesem zweiten Durchlauf hat der Client A 200 NAKs, was bedeutet, dass die Hälfte der Frames verloren gegangen sind. Dies bedeutet, dass die Geschwindigkeit des Client A 64 Kbps (d. h. ((400-200)/400)·128 Kbps = 64 Kbps) ist. Die Clients B bis E zeigen keinen Frame-Verlust im zweiten Durchlauf und folglich beträgt die Geschwindigkeit von jedem

dieser Clients mindestens 128 Kbps.

[0111] Im dritten Durchlauf ist die Geschwindigkeit der Übertragung 256 Kbps und die Clients A und B zeigen 300 bzw. 50 verlorene Frames. Von diesem dritten Durchlauf ist die Geschwindigkeit des Client A folglich 64 Kbps (d. h. ((400-300)/400)·256 Kbps = 64 Kbps), was die Messung vom zweiten Durchlauf bestätigt. Im dritten Durchlauf ist die Geschwindigkeit des Client B auch 128 Kbps (d. h. ((400-200)/400= * 256 Kbps = 128 Kbps). Die Clients C bis E haben keine Fehler in diesem dritten Durchlauf und folglich arbeiten sie jeweils mindestens so schnell wie 256 Kbps.

[0112] Im vierten Durchlauf ist die Geschwindigkeit der Übertragung 512 Kbps. Der Client A zeigt 350 verlorene Frames, misst somit (400-350)/400·512 Kbps oder 64 Kbps, was mit den vorherigen Messungen übereinstimmt. Der Client B zeigt 300 verlorene Frames, was ((400-300/400)·512 Kbps oder 128 Kbps misst, was auch mit vorherigen Durchläufen übereinstimmt. Der Client C zeigt 200 verlorene Frames, was ((400-200)/400)·512 Kbps oder 256 Kbps misst.

[0113] Im fünften Durchlauf ist die Geschwindigkeit der Übertragung 1024 Kbps. Der Client A zeigt 375 verlorene Frames, was sich auf ((400-375)/400)·1024 oder 64 Kbps beläuft, wie vorher. Der Client B misst ((400-350)/400·1024 Kbps oder 128 Kbps und der Client C misst ((400-300)/400·1024 oder 256 Kbps. Der Client D misst ((400-200)/400)·1024 oder 512 Kbps. Der Client E hat keine Ausfälle, was bedeutet, dass seine Geschwindigkeit mindestens 1024 Kbps ist.

[0114] Für jeden der fünf Durchläufe wird somit die Kapazität einer gegebenen Verbindung durch die folgende Gleichung gegeben:

((# gesandte Frames – # Naks)/# gesandte Frames)·Übertragungsgeschwindigkeit = Verbindungskapazität

[0115] Dieses Testverfahren berücksichtigt auch den Verkehr auf der Verbindung. Wenn beispielsweise eine physikalische Verbindung 256 Kbps ist und 128 Kbps Verkehr auf der Verbindung bestehen, wenn der Test durchgeführt wird, beläuft sich die Messung auf eine Kapazität von 128 Kbps, die restliche Kapazität, wenn der Verkehr berücksichtigt wird.

[0116] Die Software zum Implementieren dieser Tests kann auch verwendet werden, um die Qualität von Verbindungen zu testen, vorausgesetzt, dass die Quelle die Verbindungsgeschwindigkeiten zu jedem Client kennt. In Fig. 6 können die Verbindungsgeschwindigkeiten beispielsweise bekannt sein und es ist erwünscht, die Verbindungen mit relativ langen Testmustern zu testen, um die Framefehlerraten zu bestimmen. Eine Testdatei mit 100000 Frames könnte beispielsweise mit 64 Kbps zu der Gruppe gesandt werden, die aus den Mitgliedern A bis E besteht. Die Übertragungsrate und die NAKs werden an der Quelle gespeichert und die Anzahl von NAKs von jedem Client gibt ein Maß der Qualität (d. h. der Framefehlerrate) jeder Verbindung. Es könnte erwartet werden, dass A die schlechteste Qualität hätte, da es die am stärksten belastete Verbindung ist, und E der beste wäre, da er am wenigsten belastet ist. Andere Faktoren könnten jedoch andere Ergebnisse verursachen. Ebenso können die Geschwindigkeiten erhöht werden und überlastete Verbindungen können aus der Gruppe gelöscht werden, um die Verbindungen mit höherer Geschwindigkeit stärker zu beanspruchen.

[0117] Unter Verwendung des "Multicast-Netzwerksonden"-Merkmals der Erfindung kann folglich die Kapazität von einzelnen Verbindungen schnell durch den Server gemessen werden, wenn die einzelnen Verbindungskapazitäten unbekannt sind. Wenn die Verbindungsgeschwindigkeiten dem Server bekannt sind, kann dieses Merkmal der Erfindung auch verwendet werden, um die Qualität jeder Verbindung zu bestimmen (d. h. die Framefehlerrate jeder Verbindung zu bestimmen).

[0118] Gemäß diesem Merkmal der Erfindung wird die Konnektivität der Mitglieder der Multicast-Gruppe zuerst bestimmt, indem eine vorher beschriebene ANKÜNDIGUNGS/REGISTRIERUNGS-Phase durchlaufen wird. Das heißt, der anfängliche Schritt besteht darin, festzustellen, welche Mitglieder der Gruppe mit dem Server verbunden sind. Sobald die verbundenen Mitglieder bekannt sind, kann die Testdateiübertragung beginnen, um die Verbindungsgeschwindigkeit oder die Qualität durch den Server, der eine Testdatei zu jedem Mitglied sendet und die Ergebnisse aufzeichnet (d. h. die Anzahl von negativen Bestätigungen für jedes Gruppenmitglied), zu bestimmen.

GESCHWINDIGKEITSGRUPPEN

[0119] Mit der Kenntnis der Kapazität, Geschwindigkeit oder Bandbreite von jeder der verschiedenen Verbin-

dungen, die den Server mit den Clients koppeln (beispielsweise durch das im vorangehenden Abschnitt beschriebene "Multicast-Netzwerksonden"-Merkmal zur Verfügung gestellt), kann eine Liste dieser Geschwindigkeiten vom Server gespeichert werden. Die Liste kann dann verwendet werden, um eine Vielzahl von Client-Gruppen auf der Basis der Verbindungsgeschwindigkeit zu erzeugen oder definieren. Es können beispielsweise zwei Geschwindigkeitsgruppen vorliegen, wobei eine einen Client umfasst, der mit dem Server über eine Verbindung (oder effektive Verbindung) mit einer maximalen möglichen Geschwindigkeit von 64 Kbps verbunden ist, und wobei die andere einen Client umfasst, der mit dem Server über eine Verbindung (oder effektive Verbindung) mit einer maximalen möglichen Geschwindigkeit von 1024 Kbps verbunden ist. Die zweite Gruppe ist folglich viel schneller als die erste Gruppe. In welcher Geschwindigkeitsgruppe sich ein spezieller Rezipient befindet, beeinflusst die Übertragung von Daten zu diesem Rezipienten. Während des anfänglichen Durchgangs der Datenübertragung gemäß der Erfindung werden jedem der Rezipienten in der zweiten, schnelleren Geschwindigkeitsgruppe alle Frames vom Server gesandt und jedem der Rezipienten in der ersten, langsameren Gruppe wird nur jeder sechzehnte Frame (1/16), der zur zweiten Gruppe gesandt wurde, gesandt. Dies bedeutet, dass nach dem ersten Durchgang der Server alle Frames zu den Rezipienten der zweiten Gruppe gesandt hat, aber er nur ein Sechzehntel der Gesamtzahl von Frames zur ersten Gruppe gesandt hat. Der restliche Teil der Frames, die noch nicht zur ersten Gruppe gesandt wurden (d. h. 15/16 der Frames), wird dann zu den Rezipienten der ersten Gruppe bei nachfolgenden Durchgängen gesandt. Der Punkt besteht darin, dass, sobald der Server die Kapazität jedes Mitgliedes einer Gruppe kennt, der Server die Datenübertragung zuschneiden kann, um die Verbindungen mit höherer Kapazität zu nutzen und die Übertragung von Daten zu diesen nicht zu verlangsamen.

SAMMLUNG VON NEGATIVEN BESTÄTIGUNGEN

[0120] Wie vorstehend vorher erwähnt, kann die Anzahl von Clients, die eine Datei gemäß der Erfindung empfangen können, eine Zahl in den Tausenden sein. Folglich kann die Anzahl von Einträgen in der vom Server gewarteten Client-Statusliste eine Zahl in den Tausenden sein. Die Dateiübertragung gemäß der Erfindung kann skalierbarer gemacht werden. Sie kann beispielsweise skaliert werden, um eine Datei zu Millionen von Rezipienten/Clients anstatt Tausenden von Rezipienten/Clients zu senden. In einer bevorzugten Ausführungsform sind diese Clients oder Rezipienten Mitglieder einer Multicast-Gruppe von Clients.

[0121] Das Skalierungsmerkmal ist hilfreich, um ein potentielles Problem zu vermeiden, wenn die Anzahl von Clients in der Gruppe zu groß wird. Das Problem besteht, wenn eine große Anzahl von Clients negative Bestätigungen zum Dateisender (z. B. Server) zurücksendet und effektiv den Sender mit mehr negativen Bestätigungen drosselt als er in einem vernünftigen Zeitraum bearbeiten kann. Dies verursacht, dass die Leistung des Senders abfällt, da er eine signifikante Menge an Zeit mit dem Empfangen und Verarbeiten der negativen Bestätigungen verbringen muss und nicht für seine anderen Pflichten aufwenden kann. Dies verstopft auch die Verbindung zum Sender zurück, so dass sie mit dem Verkehr dieser negativen Bestätigungen gestaut wird.

[0122] Die Lösung für das Problem ist die "Sammlung von negativen Bestätigungen", die wiederum ermöglicht, dass die Anzahl von Client-Rezipienten drastisch von Tausenden auf Millionen erhöht wird, ohne am Dateisender/Server **20** zu verstopfen. Mit diesem Sammelmerkmal wirken bestimmte Clients oder andere Netzwerkknoten als "Weiterverbreitungspunkte" und sammeln negative Blockbestätigungen von anderen Clients. In einer bevorzugten Ausführungsform sind diese Weiterverbreitungspunkte (RPs) Router. Mit Bezug auf Fig. 7 sind fünf RPs über die Vereinigten Staaten angegeben und die Linien, die von jedem RP ausgehen, stellen den einen oder die mehreren Clients, die mit diesem RP verbunden sind, dar. Der RP **100** hat beispielsweise 1200 Clients darunter, der RP **102** hat 900 Clients, der RP **104** hat 100 Clients, der RP **106** hat 800 Clients und der RP **108** hat 500 Clients. Der Server oder die Quelle **20** befindet sich an einer anderen Stelle in den Vereinigten Staaten. Der RP **100** sammelt alle negativen Blockbestätigungen von den (z. B. 1200) Clients, die diesem zugeordnet sind oder mit diesem verbunden sind. Die anderen RPs **102**, **104**, **106**, **108** tun dasselbe für ihre zugeordneten Clients. Für jeden RP sendet dieser RP, nachdem er alle negativen Blockbestätigungen von allen seinen zugehörigen Clients gesammelt hat, nur eine Bestätigungsmeldung zum Server **20** oder zu einem anderen RP in der Kette, die zum Server **20** führt. Diese eine Meldung umfasst alle negativen Blockbestätigungen von allen diesem RP zugeordneten Clients. Wenn der Server **20** schließlich diese gesammelten negativen Blockbestätigungsmeldungen von den RPs empfängt, sendet er beim nächsten Durchlauf alle negativ bestätigen Frames wieder aus. Die RPs sind für das Empfangen dieser Frames des nachfolgenden Durchgangs und das Weiterleiten derselben zu den geeigneten Clients oder einem anderen RP in der Kette verantwortlich, der sie dann zu den geeigneten Clients oder einem anderen RP in der Kette weiterleitet usw.

[0123] Variationen, Modifikationen und andere Implementierungen dessen, was hierin beschrieben ist, kommen üblichen Fachleuten ohne Abweichung vom Schutzbereich der Erfindung, wie beansprucht, in den Sinn.

Folglich soll die Erfindung nicht durch die vorangehende erläuternde Beschreibung, sondern stattdessen durch die folgenden Ansprüche definiert sein.

Patentansprüche

1. Verfahren zur Bestimmung der Kapazitäten oder Framefehlerraten von Kommunikationsverbindungen, die Rezipienten mit einer Quelle verbinden, mit den folgenden Schritten:
(A) Bestimmung sämtlicher Rezipienten, die durch die Kommunikationsverbindungen mit der Quelle verbunden sind, wobei jeder Rezipient über eine verschiedene bzw. unterschiedliche der Kommunikationsverbindungen mit der Quelle verbunden ist;
(B) Übertragung einer Anzahl von Frames von Daten von der Quelle zu den in Schritt (A) bestimmten Rezipienten mit einer vorbestimmten Rate bzw. Frequenz über die Kommunikationsverbindungen, bis sämtliche der Anzahl der Frames übertragen worden sind;
(C) Während der Durchführung des Schrittes (B) Empfang von Bestätigungen von den in Schritt (A) bestimmten Rezipienten, wobei die Bestätigungen Angaben bezüglich der Frames umfassen, die eine erneute Übertragung benötigen;
(D) Speichern der Bestätigungen und der vorbestimmten Rate bzw. Frequenz an der Quelle; und
(E) Bestimmung der Kapazität oder Framefehlerrate von einer oder mehreren der Kommunikationsverbindungen aus Informationen, die an der Quelle gespeichert sind.

2. Verfahren nach Anspruch 1, bei dem die Schritte (B), (C) und (D) für eine vorbestimmte Rate bzw. Frequenz wiederholt werden, bis die Schritte (B), (C) und (D) eine vorbestimmte Anzahl von Malen wiederholt worden sind; und die bestimmte Information eine Kapazität von einer oder mehreren der Kommunikationsverbindungen ist.

3. Verfahren nach Anspruch 2, ferner mit den folgenden Schritten:
Übertragung einer Anzahl von Frames von anderen Daten von der Quelle zu wenigstens einem der Rezipienten, der über eine der Kommunikationsverbindungen, von denen bestimmt wurde, dass sie eine erste Kapazität aufweisen, mit der Quelle verbunden ist; und
Übertragung einer Untermenge aus der Anzahl der Frames von anderen Daten von der Quelle zu wenigstens einem anderen der Rezipienten, der mit der Quelle über eine der Kommunikationsverbindungen verbunden ist, von denen bestimmt wurde, dass sie eine zweite Kapazität aufweisen, wobei die erste Kapazität größer bzw. höher als die zweite Kapazität ist.

4. Verfahren nach einem der vorstehenden Ansprüche, bei dem der Schritt (A) ausgeführt wird indem die Quelle eine Ping-Anfrage über die Kommunikationsverbindungen an die Rezipienten sendet, und die Quelle Antworten von den Rezipienten erhält, die mit der Quelle verbunden sind, und wobei die Rezipienten Mitglieder einer Multicast-Gruppe sind.

5. Verfahren nach Anspruch 4, ferner mit dem Schritt des Bestimmens an der Quelle, nachdem die Antworten von den Rezipienten erhalten worden sind, der Umlaufverzögerung, die die Ping-Anfrage benötigt, um zu jedem der Mitglieder der Multicast-Gruppe und dann zurück an die Quelle übertragen zu werden.

6. Verfahren nach einem der vorstehenden Ansprüche, bei dem die Übertragung der Frames über die Verbindung die Übertragung der Frames über ein Computernetzwerk umfasst.

7. Verfahren nach Anspruch 6, bei dem die Übertragung der Frames über das Computernetzwerk die Übertragung der Frames über das Internet umfasst.

8. Verfahren nach einem der vorstehenden Ansprüche, bei dem die Übertragung der Frames über die Verbindung die Übertragung der Frames über ein drahtloses Netzwerk umfasst.

9. Verfahren nach Anspruch 8, bei dem die Übertragung der Frames über das drahtlose Netzwerk die Übertragung der Frames über ein zellenartiges bzw. netzförmiges Netzwerk bzw. ein Mobilfunknetz umfasst.

10. Verfahren nach einem der vorstehenden Ansprüche, bei dem das Verfahren ausgeführt wird durch eine Quelle, die über die Verbindung mit den Rezipienten verbunden ist, und wobei eine erste Quellen-Rezipienten-Übertragungsverzögerung mit einem der Rezipienten assoziiert wird, und eine zweite, unterschiedliche Quellen-Rezipienten-Übertragungsverzögerung mit einen weiteren der Rezipienten assoziiert wird.

11. Verfahren nach einem der vorstehenden Ansprüche, bei dem Schritt (B) die Übertragung der Daten beinhaltet, welche eine Computerdatei umfassen.

12. Verfahren nach einem der vorstehenden Ansprüche, bei dem Schritt (B) eine Multicast-Übertragung der Frames über die Verbindung zu den Rezipienten umfasst.

13. Verfahren nach einem der vorstehenden Ansprüche, bei dem Schritt (C) den Empfang der Bestätigungen umfasst, welche jeweils eine negative Bestätigung umfasst bzw. umfassen, welche angibt, dass ein bestimmter der Rezipienten eine erneute Übertragung eines oder mehrerer der Frames benötigt.

Es folgen 6 Blatt Zeichnungen

Anhängende Zeichnungen

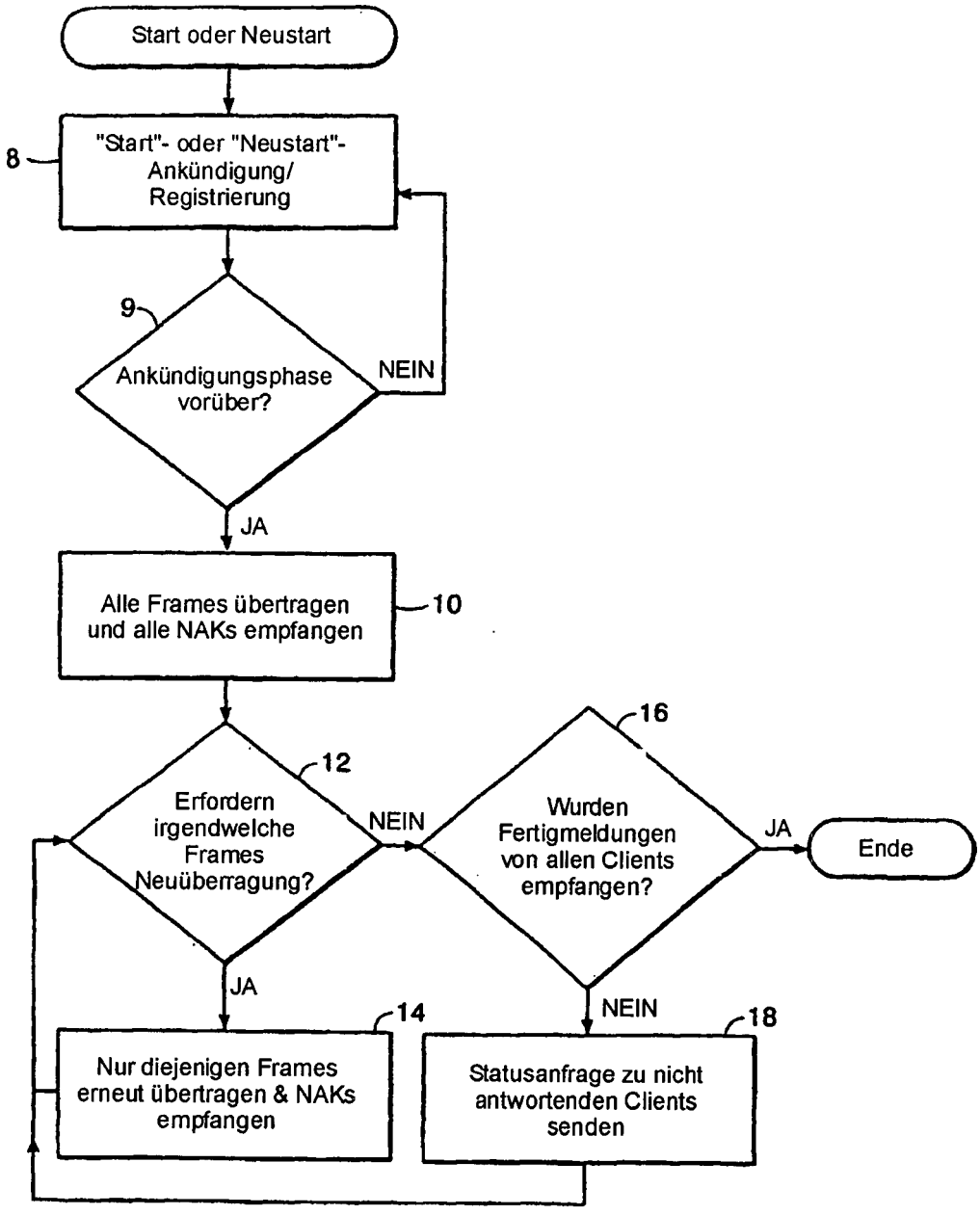


FIG. 1

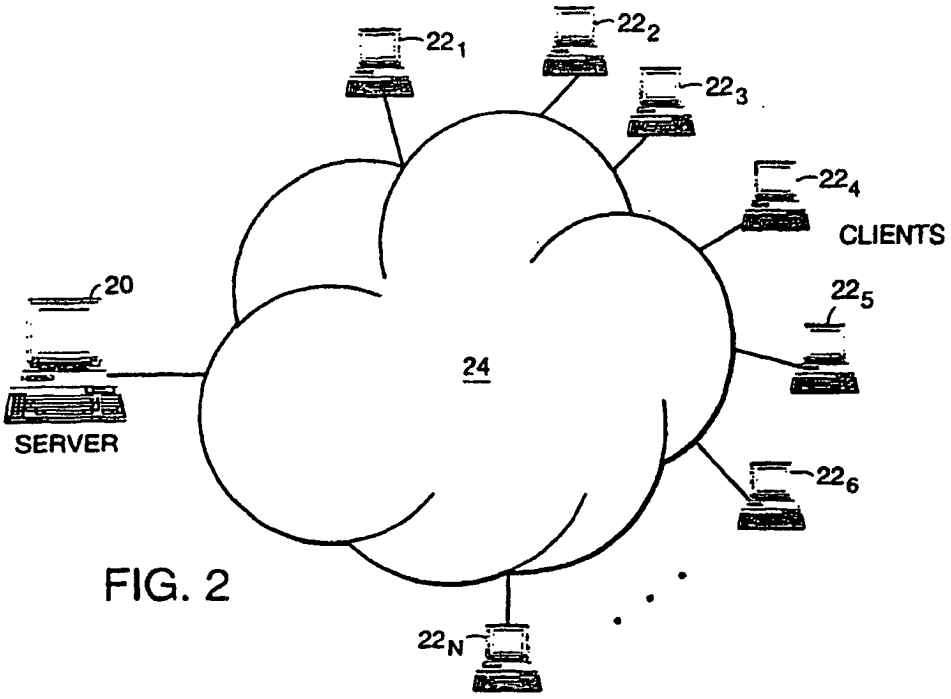

22 1
22 2
22 3
22 4
CLIENTS
22 5
22 6
22 N
20
SERVER
24

FIG. 2

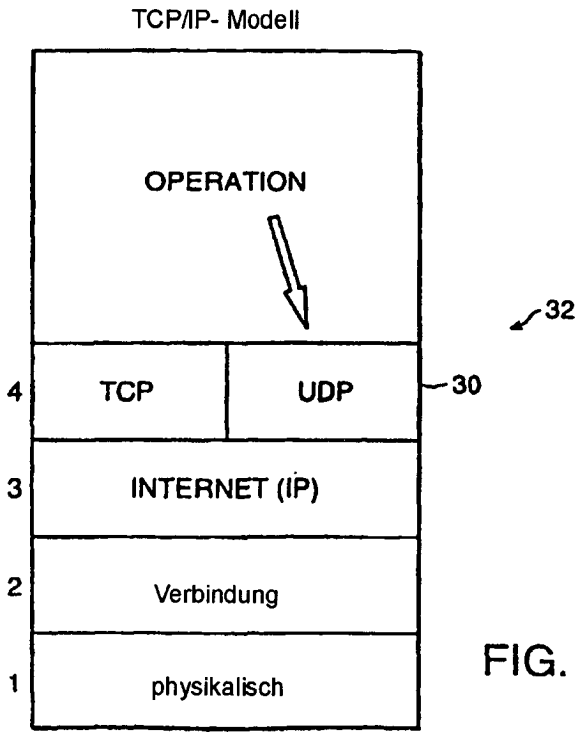

TCP/IP- Modell
OPERATION
32
4
TCP
UDP
30
3
INTERNET (IP)
2
Verbindung
1
physikalisch

FIG. 3

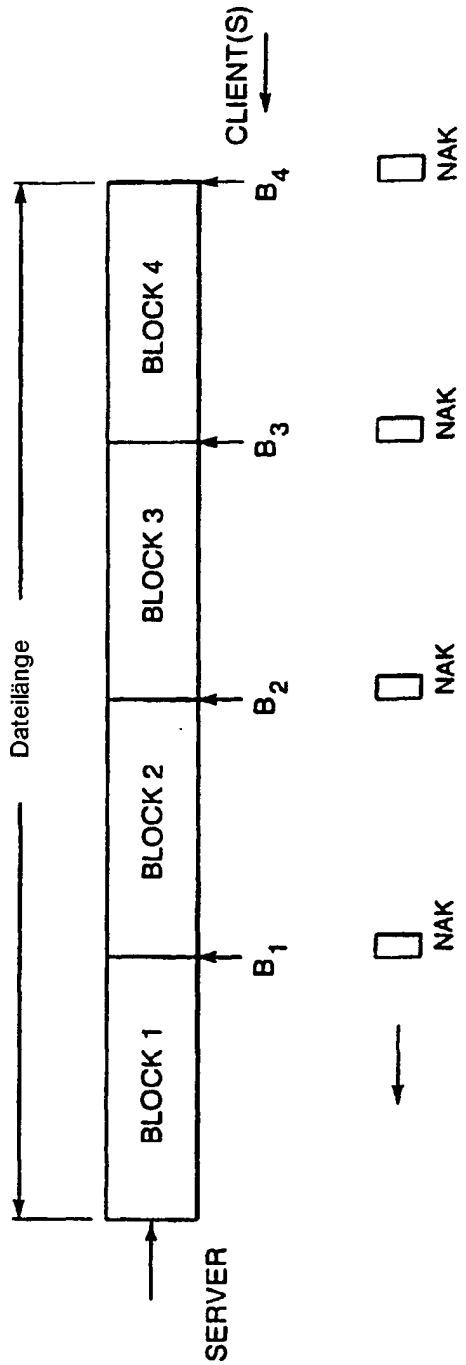

CLIENT(S)
Dateilänge
BLOCK 1
BLOCK 2
BLOCK 3
BLOCK 4
SERVER
B1
B2
B3
B4
NAK
NAK
NAK
NAK

FIG. 4

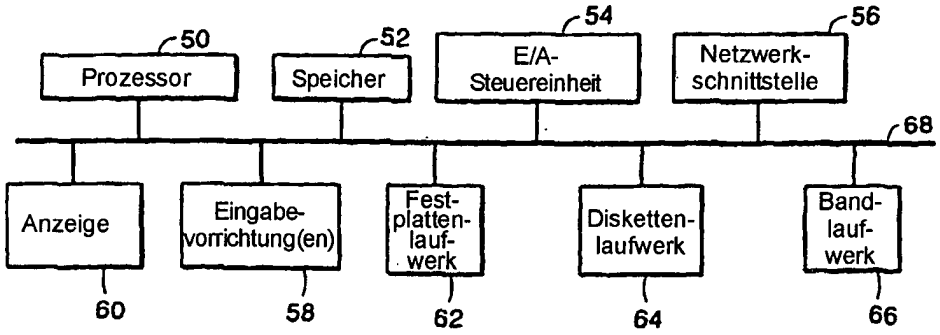
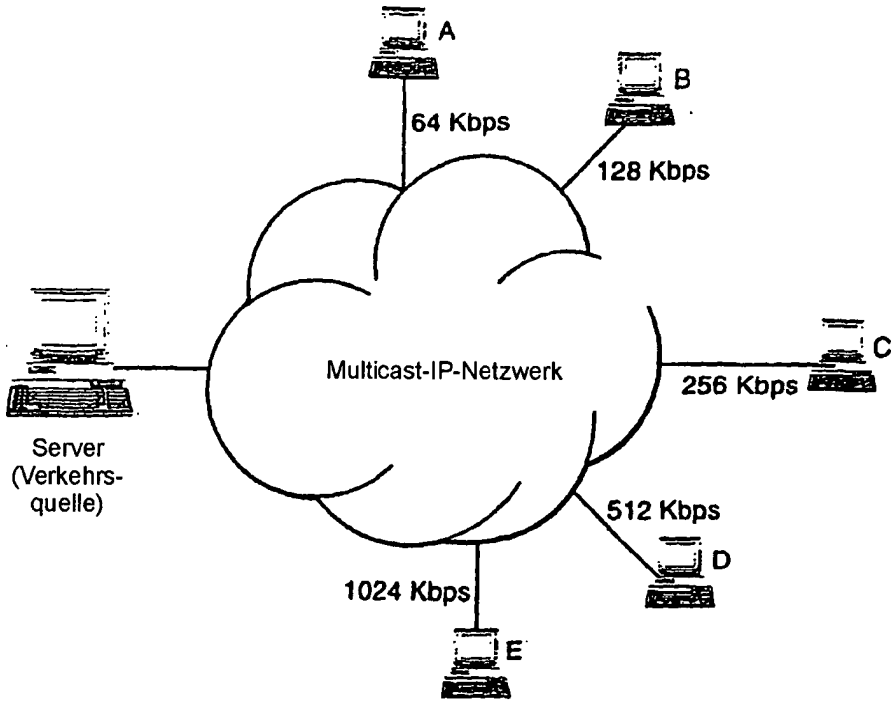

50
Prozessor
52
Speicher
54
E/A-
Steuereinheit
56
Netzwerk-
schnittstelle
68
Anzeige
Eingabe-
vorrichtung(en)
Fest-
platten-
lauf-
werk
Disketten-
laufwerk
Band-
lauf-
werk
60
58
62
64
66

FIG. 5



heterogenes Multicast-IP-Netzwerk
(z. B. MBONE)

FIG. 6

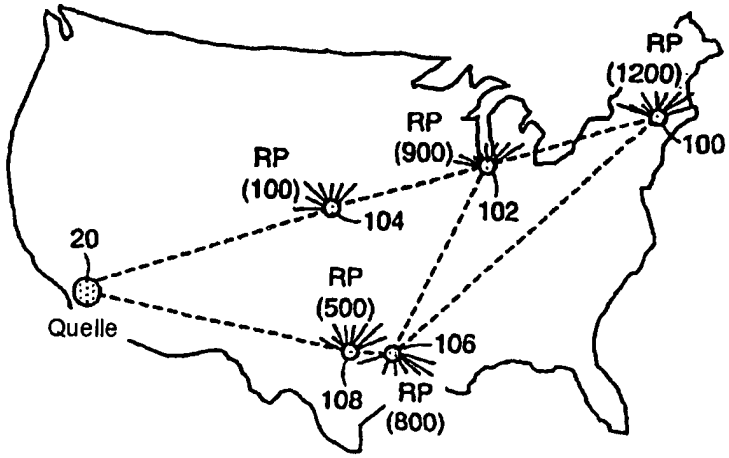


FIG. 7

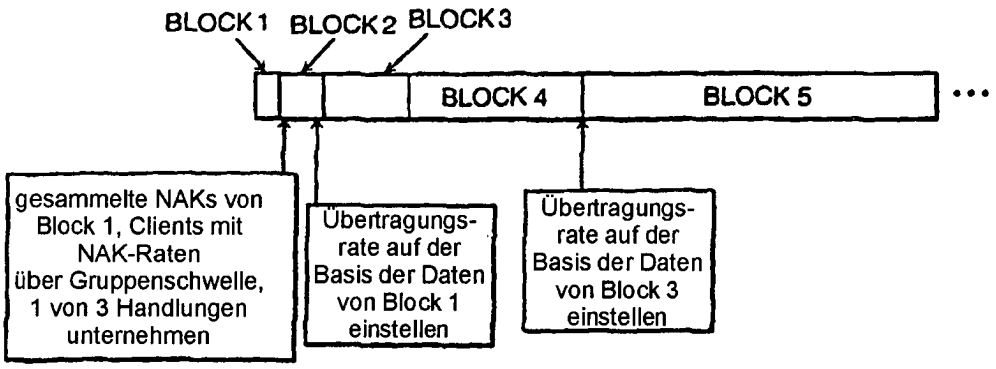

BLOCK 1
BLOCK 2
BLOCK 3
BLOCK 4
BLOCK 5
gesammelte NAKs von Block 1, Clients mit NAK-Raten über Gruppenschwelle, 1 von 3 Handlungen unternehmen
Übertragungs-rate auf der Basis der Daten von Block 1 einstellen
Übertragungs-rate auf der Basis der Daten von Block 3 einstellen

FIG. 8

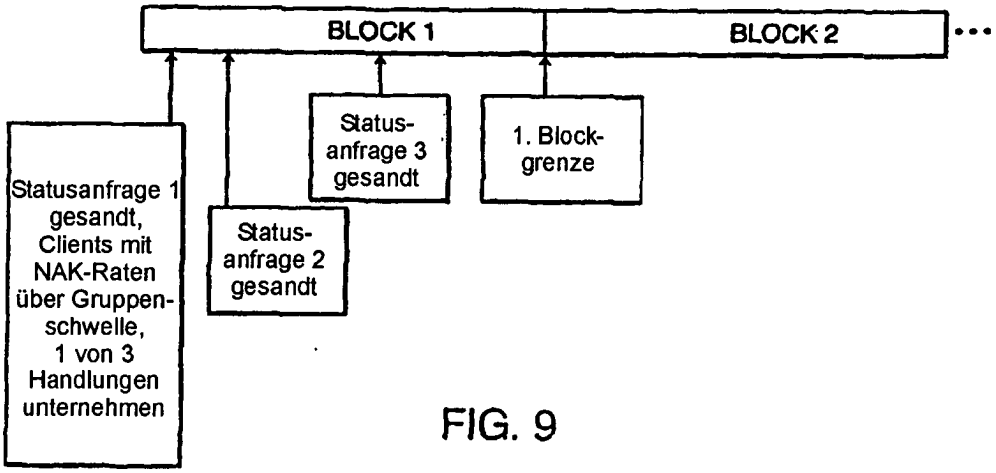

BLOCK 1
BLOCK 2
Statusanfrage 1 gesandt, Clients mit NAK-Raten über Gruppen-schwelle, 1 von 3 Handlungen unternehmen
Status-anfrage 2 gesandt
Status-anfrage 3 gesandt
1. Block-grenze

FIG. 9