(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第3690809号
(P3690809)
(45)発行日　平成17年8月31日(2005.8.31)　(24)登録日　平成17年6月24日(2005.6.24)

(51)Int.Cl.[7]	F I		
H04L 12/56	H04L	11/20	102A
G06F 13/00	G06F	13/00	351A
H04L 1/18	H04L	1/18	
H04L 12/18	H04L	11/18	
H04L 12/28	H04L	11/00	310D

請求項の数 12　(全 24 頁)

(21)出願番号　特願平8-522373
(86)(22)出願日　平成8年1月16日(1996.1.16)
(65)公表番号　特表平10-512726
(43)公表日　平成10年12月2日(1998.12.2)
(86)国際出願番号　PCT/US1996/000634
(87)国際公開番号　WO1996/022641
(87)国際公開日　平成8年7月25日(1996.7.25)
審査請求日　平成15年1月14日(2003.1.14)
(31)優先権主張番号　08/375,493
(32)優先日　平成7年1月19日(1995.1.19)
(33)優先権主張国　米国(US)

(73)特許権者
ザ　ファンタスティック　コーポレイション
スイス国　６３０１　ツーク，　ランディス　ウント　ギル－シュトラーセ　３
(74)代理人
弁理士　山本　秀策
(74)代理人
弁理士　安村　高明
(74)代理人
弁理士　森下　夏樹
(72)発明者　ミラー，シー．ケニース
アメリカ合衆国　マサチューセッツ　０１７４２，コンコード，レボリューショナリー　ロード　８５

最終頁に続く

(54)【発明の名称】不要な再送信を防止するＡＲＱ技術を用いたネットワークマルチキャスティング方法

(57)【特許請求の範囲】
【請求項１】
通信リンク上でデータを送信する方法であって、
（Ａ）該データを複数のブロックに区分するステップであって、該複数のブロックのそれぞれが複数のパケットを含み、該複数のパケットのそれぞれが１つ以上のビットを含み、該区分するステップがブロックごとのパケットの数を、パケットごとに利用できるビットの最大数とほぼ同等に設定することを含む、ステップと、
（Ｂ）該複数のブロックのそれぞれにおける該複数のパケットのすべてを、ブロックごとに、１以上の受け手に送信するステップと、
（Ｃ）送信の間に、該送信されたブロック内の再送信を要求するパケットの指示を、該１以上の受け手のうちの１以上から受信するステップであって、それぞれの指示は該１以上の受け手ごとに関連付けられており、該パケットの１つに含まれるビットマップを含む指示であって、該ビットマップが、該パケット内で利用できるビットの最大数にほぼ同等の複数のビットを含み、該ビットマップの各ビットが、該送信されるブロック内の該パケットの異なる１つを表す、ステップと、
（Ｄ）該ステップ（Ｂ）、（Ｃ）および（Ｄ）を繰り返すことにより、再送信を必要とするパケットのみを再送信するステップと、
を含む方法。
【請求項２】
前記ステップ（Ｄ）が、前記ステップ（Ｂ）、（Ｃ）および（Ｄ）を所定の回数繰り返す

ことを含む、請求項１に記載の方法。
【請求項３】
前記ステップ（Ｄ）が、前記ステップ（Ｂ）、（Ｃ）および（Ｄ）を所定の量の時間繰り返すことを含む、請求項１に記載の方法。
【請求項４】
前記通信リンクがインターネットを含む、請求項１に記載の方法。
【請求項５】
前記通信リンクがセルラーネットワークを含む、請求項１に記載の方法。
【請求項６】
前記ステップ（Ｂ）が前記複数のパケットを所定の速度で送信することをさらに含む、請求項１に記載の方法。
【請求項７】
送信を受け取っていない受信者がいる場合、どの受信者であるかを判別し、その後、該判別結果に基づき前記所定の速度を調節して、受信者のパケット受信を増大させるステップをさらに含む、請求項６に記載の方法。
【請求項８】
前記データは、１つ以上のコンピュータファイルを含んでおり、前記区分するステップは、該１つ以上のコンピュータファイルを壊して前記複数のブロックにする、請求項１に記載の方法。
【請求項９】
前記ステップ（Ｃ）において受信された指示は、前記１以上の受け手のうち特定の受け手が前記パケットのうち１つ以上のパケットの再送信を要求するという否定応答確認を含む、請求項１に記載の方法。
【請求項１０】
前記通信リンクは無線リンクを含み、前記ステップ（Ｂ）、（Ｃ）および（Ｄ）は、該無線リンクを介した送信を含む、請求項１に記載の方法。
【請求項１１】
前記区分するステップは、ブロックごとのパケット数をパケットごとに利用可能な最大ビット数に等しくなるように設定する、請求項１に記載の方法。
【請求項１２】
前記区分するステップは、ブロックごとのパケット数をパケットごとに利用可能な最大ビット数にほぼ等しくなるが、該最大ビット数未満となるように設定する、請求項１に記載の方法。
【発明の詳細な説明】
関連する出願への相互参照
本願は、1995年1月19日に出願され、本願の出願日の時点では係属中である、米国特許出願第08/375,493号（代理人番号第PSM-001号）に関連している。また、本願は、その他２つの米国特許出願にも関連している。これらその他の出願は、ともに本願と同じ日に米国特許および商標庁に出願される。これらその他の出願は、代理人番号第STR-001CP1号および第STR-001CP2号により識別される。これら２つのその他の出願および米国特許出願第08/375,493号はいずれも、本願において参考として援用される。
発明の分野
本発明は、データ送信に関する。具体的には、本発明は、サーバからクライアントへの高速で信頼性の高いファイル送信に関する。
発明の背景
ワイドエリアネットワーク（WAN）のようなコンピュータネットワークは、サーバノードおよび１以上のクライアントノードのようなネットワーク加入者の間での通信を可能にするように、ユニキャスト、マルチキャストおよびブロードキャストサービスを提供している。マルチキャストフレームリレーは、コンピュータネットワーク上で通信するために利用されるサービスである。マルチキャストIPテクノロジーもまた、コンピュータネットワ

ーク上で通信するために利用されるサービスである。ブロードキャストフレームリレーは、衛星ネットワーク上で通信するために利用されるサービスである。ここで、用語「ブロードキャスト」は、ネットワークに接続されたクライアントノードのすべてに情報を送るサーバノードを指している。用語「マルチキャスト」は、ネットワークに接続されたすべてのクライアントノード中の１サブセットに情報を送るサーバノードを指している。ブロードキャストおよびマルチキャストは、WANを用いた比較的、新しいネットワーク機能である。

情報プロバイダの中には、中央ロケーションにおけるサーバノードから遠隔カスタマロケーションにおける１以上のクライアントノードへと、サーバおよびクライアントが結合されているコンピュータネットワークを介して、情報を電子的にブロードキャストまたはマルチキャストすることにより情報を伝達したいと望んでいるものがある。ブロードキャストおよびマルチキャストネットワークは、伝達した情報についての応答確認を全く提供しないので、これらのサーバは信頼できない可能性がある。このような信頼性の低さは、一般に、情報プロバイダには望ましくなく、受け入れがたいことである。

コンピュータネットワークで通常用いられる１組のプロトコルは、TCP/IPである。これは、インターネットで用いられているプロトコルである。TCPとは、伝送制御プロトコルを表し、IPとは、インターネットプロトコルを表す。TCP/IPに関しては、２つのファイル転送プロトコルが利用可能である。すなわち、（i）TCPの最上位のアプリケーションをランするファイル転送プロトコル（FTP）と、（ii）UDPの最上位でランする簡易ファイル転送プロトコル（TFTP）との２つである。UDPとは、ユーザデータグラムプロトコルを表す。TCPおよびUDPはともに、インターネットワーク（すなわち、ネットワークのネットワーク）を介した情報のエンドツーエンド伝達の責任をもつトランスポートプロトコルである。FTPおよびTFPTはともに、ポイントツーポイント（つまり、ユニキャスト）ファイル転送のみをサポートする。FTPは、信頼性ある伝達についてはTCPに依存する。なぜなら、TCPは、コネクションオリエンテッド応答確認型トランスポートプロトコルであるからである。TFTPは、信頼性については独自の応答確認を提供する。なぜなら、TFTPは、応答確認をサポートしないコネクションレストランスポートサービスであるUDPの最上位でランするからである。

TCPのような接続指向のプロトコルは、バーチャルサーキット接続のセットアップおよびティアダウンを要求する。そのオーバーヘッドが比較的長いので、TCPおよびこれに類似したプロトコルは、セルラーディジタルパケットデータ（CDPD）ネットワークのような本質的には貧弱な接続のネットワークでは望ましくない。CDPDは、TCP/IPを、ネットワークで用いられる１次プロトコルスイートとして利用する。CDPD無線ネットワークは、UDP（コネンクションレストランスポート層）よりも上位のアプリケーションのみで動作することを勧める。よって、CDPDに選択されるファイル転送プロトコルは、TFTPとなる。

TFTPは、ファイルを、それぞれのデータについて512バイトを有する複数のパケットに分解した後、それぞれのデータパケットを１度に１個ずつ送る。それぞれのデータパケットが送られた後、TFTPは、送信側のノードに次のデータパケットの送信を許可する前に、送信側のノードに、少なくとも１つの受信側ノードからの応答確認を待たせる。TFTPは、例えば、Douglas E. Comer（Internetworking with TCP/IP, Volume I, Principles, Protocols and Architecture、第2版、Prentice Hall、1991年、第23章、第377～390頁）により著された文献に記載されている。

応答確認はTFTPの一部ではあるものの、TFTPにおいて用いられている応答確認スキームは、ネットワーク遅延が顕著になるにつれて、および／または受信側ノード中の２つ以上で異なってくると、非常に効率が悪くなる。現在知られているデータ転送メカニズムの中には、TFTPと同様に、パケット毎の応答確認を要求するものもある。よって、このようなその他のメカニズムも、データ全体の転送については、比較的低速である。

発明の要旨

本発明の目的は、通信リンク上でサーバから１つ以上のクライアントへとファイルを高速かつ信頼性よく送信することである。ファイルの転送は、好ましくは、複数のクライアン

トへのマルチキャスト送信である。概略的にいうと、本発明によるファイル転送は、たとえ遅延が顕著になっても、および／または受信側クライアント中の２つ以上で遅延が異なってきても、リンク遅延時にさえ速度、信頼性および効率の低下を被ることがない。本発明は、コンピュータソフトウェアファイルを電子的に分配するのに理想的なメカニズムを提供する。

サーバを複数のクライアントに結合し、それらの間での通信を可能にする通信リンクは、コンピュータネットワーク（例えば、LAN、WAN、インターネットなど）でも、無線ネットワーク（例えば、CDPDのようなパケットセルラーデータネットワーク）でも、これらのタイプの通信媒体の組み合わせでも、あるいは、一般に高速で遅延の少ないネットワークである、例えば衛星ネットワークのようなその他の通信媒体であってもよい。

本発明によれば、クライアントは、サーバがデータファイルを送っている間に、否定応答確認のみをサーバに送り返す。この通信は、連続的である。すなわち、サーバは、データの送信を中止してクライアントからの否定応答確認を待つのではなく、サーバは、サーバがデータを送信している間にクライアントの否定応答確認を受信する。クライアントの否定応答確認は、サーバに対して、特にどのパケットが再送信される必要があるかを指示する。パケットが再送信される必要があるのは、例えば、そのパケットが、１以上のクライアントによって受信されなかったか、あるいは誤って受信されたからである。サーバがリンクを介してデータの全体（例えば、ファイルの全体）をクライアントに送った後、サーバは、第２ラウンドの送信をおこなう。この第２ラウンドでは、サーバは、クライアントにより再送信を要求すると指示された特定のパケットのみを再送信する。この第２ラウンドの間も、クライアントは、やはり否定応答確認（すなわち、全く受信されなかったか、または正しく受信されなかったパケットの指示）のみを送る。その後、このプロセスは、それぞれのクライアントがすべてのパケットを正しく受信するのに必要であれば、さらに多数のラウンドの再送信を継続しておこなうことができる。あるいは、再送信ラウンドは、所定の回数だけ繰り返されてもよい。この回数は修正可能である（つまり、この回数は設定可能である）。後続するラウンドはそれぞれ、典型的には、その直前のラウンドよりも少数のパケットの送信を伴う。なぜなら、直前で誤りを含むパケットのみが再び送られるからである。

このスキームは、サーバから１以上のクライアントへとデータを迅速に、かつ信頼性よく転送する。このスキームが迅速なのは、パケット間の境界で停止して、今送信したばかりのパケットについてクライアントからの否定応答確認を待つことなく、ファイルの全体を転送することがサーバに許可されるからである。すなわち、データの転送は、それぞれのデータ転送ラウンドが、特定のクライアントの受信問題に関わりなく、および／またはどのようなリンク遅延の問題（例えば、パケットがサーバからあるクライアントへ到達するのに要する時間と、パケットがそのサーバから別のクライアントへと到達するのに要する時間との間の差）にも関わりなく継続するという点で、否定応答確認には直接結びつくものではない。また、後続するそれぞれの送信ラウンドは、直前のラウンドの間に受信されなかったか、または誤って受信されたパケットの送信のみを伴うので、概略的にいうと、サーバが、同一のファイルの全体を１回よりも多く送ることが必要になることは決してない。このスキームの信頼性が高いのは、それぞれのクライアントにあらゆるパケットを提供しようとしているからであり、また、概略的にいうと、特定個人のクライアントの受信問題が、その他のクライアントの受信速度および精度に悪影響を及ぼすことがないからである。

本発明によるデータ転送は、クライアントの誰からも肯定応答確認を要求することはないし、また期待することもない。もし否定応答確認がサーバに戻ってきて受信されなければ、肯定応答確認が暗黙のうちに示される。また、本発明によれば、好ましくは、複数の否定応答確認が収集され、「マルチプルセレクティブリジェクト否定応答確認」としてサーバに送り返される。典型的には、少なくとも１つのこのようなマルチプルセレクティブリジェクト否定応答確認が、例えば、サーバからクライアントへの第１ラウンドの送信の間にサーバに送り返される。１つのマルチプルセレクティブリジェクト否定応答確認が、何

百もの個別否定応答確認を表現することができる。これらの否定応答確認の集合を用いることによって、リンク上のトラヒックを大幅に減らすことができ、サーバからクライアントへのデータ転送あるいはその他の目的のためにリンク上の帯域幅を解放することができる。本発明では、概略的にいうと、サーバとリンクとが、同時に、またはごく短い時間帯の間にまとめて戻ってくる多数の個別否定応答確認で閉塞されることはない。このように、リンク上でサーバへと送られつつある個別応答確認の個数を減らすことができる結果、スケーラビリティを改善できるという効果が得られ、有意な利点が得られる。すなわち、マルチプルセレクティブリジェクト否定応答確認を使用すれば、１個のファイルを送ることができるクライアントの数は、サーバに戻ってくる応答確認トラヒックが減るので増やすことができる。

本発明の好ましい実施の形態では、転送されるべきデータの全体（例えば、１個のファイル）は、複数のブロックに分割される。ここで、それぞれのブロックは複数のパケットを有する。サーバは、すべてのブロック（例えば、１個のファイル全体）の送信を終了すると、１ラウンドを完了する。１個の完全なブロックが送信された後、クライアントは、戻りユニキャスト通信パスを介して、否定応答確認をサーバに送り返す。ブロックの境界は、クライアントによる否定応答確認の送信をトリガする。ブロックNについて、否定応答確認がクライアントからサーバへと戻ってくる時、サーバは、ブロックN＋1（つまり、後続ブロック）をクライアントに送出しつつあるか、または、サーバは、既にすべてのブロックの送信を終了している。

本発明によれば、以下の特徴が提供される。まず、伝送レートを設定し、マルチキャストグループを規定する能力が得られる。また、「マルチキャストネットワークプローブ」という特徴を用いて、容量が未知であるリンクについてその容量を決定し、同一の特徴を用いることによって、容量が既知であるリンクのフレームエラーレートを決定することができる。「マルチキャストピン」という特徴は、ソースと、マルチキャストグループのメンバーたちとの間のコネクティビティを決定するのに用いることができる。「シードグループ」は、リンクの容量を求めた後にセットアップすることができる。あるいは、もしその容量が分かっているのなら、第１のパスでは、最も高速のリンクによってソースに接続されている受け手はすべてのデータを受信するようにし、より低速なリンクの受け手は、そのデータの一部のみを受信するように、セットアップすることもできる。ソースからデータを受信できる受け手の数は、「否定応答確認収集」スキームを用いることによって、大幅に（例えば、1000倍以上に）増やすことができる。これにより、「複製点（replication point）」（好ましくは、ルータ）は、個々の否定応答確認を収集し、それらを１単位として次のレベルに送り出すことができる。

なお、用語「パケット」、「データグラム」および「フレーム」は、本願明細書においては、同一のものを識別するために用いられており、互いに置き換え可能な用語である。すなわち、これらの用語は、ソースアドレスおよびデスティネーションアドレスをその一部としてもつことができ、リンクを介して送信されるデータまたは情報の単位を指す言葉である。

本発明の上記目的、局面、特徴および利点、ならびに、その他の目的、局面、特徴および利点は、以下の説明および請求の範囲から明らかになるであろう。

【図面の簡単な説明】

図面において、同一の参照番号は、すべての図を通して広く同一の部分を指している。また、これらの図面は必ずしも現実の縮尺に即したものではなく、概略的には、本発明の原理を説明するために強調されているところもある。

図1は、本発明によるデータ送信動作のフローチャートである。

図2は、サーバが１以上のクライアントと通信することができるようにする物理構成の図である。

図3は、TCP/IPプロトコルスタックに対する本発明の一実施の形態のロケーションを示す図である。

図4は、本発明による、「第１パス」のブロックおよびフレーム送信および応答確認プロ

セスの図である。
図5は、本発明の少なくとも一部が実施可能である、サーバの簡略ブロック図である。
図6は、マルチキャストグループのメンバーたちが別々の容量のリンクにより接続されている、混成マルチキャストネットワークの図である。
図7は、スケーラビリティを向上させ、何百万もの受け手が送り手からのデータを迅速に、かつ信頼性よく受信できるようにする、本発明による応答確認収集という特徴を説明する図である。
図8は、可変ブロックサイズ方法を用いる、輻輳／フロー制御に関連する図である。
図9は、ブロック境界よりも前にクライアントからの否定応答確認を要請する好ましいステータスリクエスト方法を用いる、輻輳／フロー制御に関連する図である。

説明

図1および図2を参照すると、本発明によれば、通信リンク24上でのソースつまりサーバ20から１以上の受け手つまりレシーバつまりクライアント22_1、22_2、…、22_Nへの迅速で、信頼性の高いデータ送信は、複数のフレームのかたちのデータ（例えば、ファイル）を、そのファイルの全体（すなわち、それら複数のフレームのすべて）がリンク24上で送信されてしまうまで、リンク24上で１以上の受け手22に送信すること（ステップ10）を含む。これらのフレームが送信されている間に、１以上の受け手22からのフレーム否定応答確認がリンク24を介して受信される（ステップ10）。もし、ファイルの全体がリンク24上で送信された後に、いくつかのフレームがリンク24上で再送信される必要があると否定応答確認が指示すれば（ステップ12）、これらいくつかのフレームのみが再送信される（ステップ14）。これらいくかのフレームがリンク24上で再送信されている間に、１以上の受け手22からのフレーム否定応答確認がリンク24を介して受信される（ステップ14）。このプロセスは、その後、ステップ12、14および16によって示されるフレーム再送信の要求がなくなるまで、必要な回数だけ繰り返される。ステップ16では、サーバ20は、すべての受け手22による「ダン（done）」メッセージがサーバ20において受信されたかどうかを判定する。もし受け手が「ダン」したのなら、その受け手は、すべてのフレームを既に受信しており、サーバ20に対して「ダン」メッセージを既に送ってその旨を示していることになる。「ダン」である受け手たちは、自分の名前が「ダンリスト」に載っているのを見るまで「ダン」メッセージをサーバに送り続ける。この「ダンリスト」は、すべての「ダン」である受け手（つまり、「ダンリスト」に載っている受け手）に対して、「ダン」メッセージをサーバに送ることをやめることを指示する通知としてサーバが送るものである。所定の時間の経過後、または所定のイベントの後に、サーバ20は、応答しないすべての受け手22（すなわち、サーバが、「ダン」メッセージを受信していない受け手）に対してステータスリクエストを送る（ステップ18）。ファイル全体の初期転送および後続するエラーフレームの一回の送信を、ここでは広く「ラウンド」または「パス」と称することにする。
第１のパスでは、サーバ20は、好ましくは、ファイルをすべてのクライアント22中の１サブセットにマルチキャストする。典型的には、これらのクライアント22のうち少なくとも２つで、それぞれに伴うサーバ・クライアント間フレーム伝送遅延が異なってくる。本発明によるデータ送信は、たとえ遅延の差が顕著であっても、また、たとえあらゆるクライアント22でそれぞれに伴う遅延が異なっているとしても、このような遅延の差に影響されることはない。
リンク24は、コンピュータネットワーク（例えば、LAN、WAN、インターネットなど）でも、無線ネットワーク（例えば、セルラーデータネットワーク）でも、これら２つのタイプの通信媒体の組み合わせでも、あるいは、典型的には、高速で遅延の少ない、例えば衛星ネットワークのようなその他の通信媒体であってもよい。第１ラウンドの間にリンク24上で送信される複数のフレームは、それら全部を合わせて、サーバ20から１以上のクライアント22に転送されている、１個のコンピュータデータファイルを表すことができる。
サーバ20およびクライアント22は、DOSを含む多種多様なオペレーティングシステムのいずれか１つをランする、PCあるいはワークステーションのようなコンピュータでありうる。図5を参照すると、サーバ20は、どのタイプのコンピュータであるかには関わりなく、

典型的には、中央プロセッサ50と、プログラムおよび／またはデータを格納するメインメモリユニット52と、入力／出力コントローラ54と、ネットワークインタフェース56と、キーボートやマウスのような１つ以上の入力デバイス58と、ディスプレイデバイス60と、固定されたドライブユニット、すなわちハードディスクドライブユニット62と、フロッピーディスクドライブユニット64と、テープドライブユニット66と、これらの機器間での通信を可能にするようにこれらの機器を結合するデータバス68と、を備えている。クライアントコンピュータ22はそれぞれ、一般に、図5のサーバ20に含まれている機器のすべて、またはその一部を備えている。

ある実施の形態では、１つ以上のコンピュータプログラムが、サーバ20およびクライアント22の演算能力を規定する。これらのプログラムは、ハードドライブ62、フロッピードライブ64および／またはテープドライブ66を介してサーバ20およびクライアント22にロードされうる。あるいは、これらのプログラムは、メインメモリ52の永久記憶部分（例えば、ROMチップ）に常駐していてもよい。他の実施の形態では、サーバ20および／またはクライアント22は、コンピュータプログラムからインストラクションを受けることを必要とせずにここで記載されるすべての機能を実行する、特別に設計され、専用で、結線された電子回路を備えていてもよい。本発明は、例えば、クライアントソフトウェアの新しい改版レベルをサーバから１以上のクライアントへと電子的に、迅速かつ高い信頼性でロードするのに用いることができる。

図3を参照すると、本発明は、好ましくは、TCP/IPプロトコルスタック32のUDPよりも上位にあるアプリケーション層30で動作する。また、本発明は、ネットウェアSPX/IPXプロトコルスイートにおけるIPXのようなその他のプロトコルスタックに存在するコネクションレストランスポート層の上層にあるアプリケーション層において動作することもできる。UDPは、ユーザデータグラムプロトコルを表しており、また、あるコンピュータ上のアプリケーションプログラムが、別のコンピュータ上のアプリケーションプログラムへとデータグラムを送ることができるようにするのは、TCP/IP標準プロトコルである。UDPは、データグラムを伝達するためにインターネットプロトコル（IP）を用いる。UDPデータグラムがIPデータグラムと異なるのは、ダイアグラムの送り手に、受信側コンピュータ上の多数のデスティネーション（つまり、アプリケーションプログラム）からの識別を可能にする、プロトコルポートナンバをUDPデータグラムが含んでいることである。UDPデータグラムは、また、典型的には、送られているデータに対するチェックサムを含んでいる。

広くいうと、本発明によるデータ送信は、４つの局面、すなわち「アイドル」、「アナウンス／登録」、「転送」および「完了」を含んでいる。「アイドル」状態では、活動はおこなわれない。データの集合体（例えば、ファイル）がサーバ20により送信用に選択されると、「アナウンス／登録」フェーズに入る。これら４つのフェーズのいずれの間でも、サーバ20におけるオペレータにはすべてのファイルが利用可能である。

アナウンス／登録

このフェーズ（図1のステップ8）では、サーバは、クライアントに対して、１個のファイルがまさに転送されつつあることを「アナウンス」し、そのファイルの転送に伴うパラメータを提供する。このフェーズの最大持続時間は、分単位で表され、設定可能である。「アナウンス」メッセージは、マルチキャストグループをセットアップするのに用いられ、クラスDのアドレスは、マルチキャストグループの割り当てに用いられる。

クライアントは、「アナウンス」メッセージを受信したことをサーバに登録することを強制される。クライアントはこの「アナウンス」メッセージを見ると、自分が、そのメッセージ中で識別されているグループに関連づけられていることを確認する。レシーバが正しいサーバIPアドレスと、正しいポートナンバとをもっていることは、「アナウンス」メッセージを処理できるレシーバには暗黙のうちに示される。クライアントは、自分のアドレスが後続する「アナウンス」パケット内の登録されたクライアントリスト内にあるのを見るまで、「アナウンス」パケットに対して「登録」パケットを用いて自動的に応答する。「登録」パケットは、クライアントの加入に関するサーバへの肯定応答確認として作用する。いったんサーバがクライアントの「登録」パケットを受信すると、サーバは、そのク

ライアントを「アナウンス」パケットの次のブロードキャストにおけるクライアントリストに加える。クライアントリストは、サーバにより保守される。クライアントが、そのクライアントのIDがクライアントリスト内に含まれている「アナウンス」パケットを受信すると、そのクライアントの登録は完了する。期待されたすべてのレシーバが「アナウンス」メッセージに応答した時、または、「アナウンス」タイムアウトが時間切れになった時、これらのどちらが最初に起こっても、その時点でファイルの実際の転送が始まる。クライアントには、送られようとしているファイルを操作するリソースがあるので、このクライアントはそのグループに加入できることを、この登録は示している。望まざる加入を防止するために、グループセットアップ時に暗号鍵交換をおこなうことができる。いったんファイルの転送が始まると、「アナウンス」パケットは送られなくなり、「アナウンス」フェーズは、終了する（図1のステップ9）。

ファイル送信のすべての特性は、「アナウンス」パケットに入れて送信される。この「アナウンス」メッセージを受信するとただちに、クライアントは、ユニキャストデータグラムを用いてサーバに応答する。この応答は、レシーバがファイルを受信するための設備をもっているかどうかを示す。また、この応答は、送信アボートの場合には、クライアントが送信を続行（図1に示されている「再開」）するのに十分なコンテクストをもっているかどうかも示す。いくつかの事例でのアナウンス期間の持続時間は、サーバサイトのオペレータが、クライアントサイトに対する呼を開始し、コンピュータが利用可能ではないか、または転送用の設備をもっていないことを示すことができるだけの長さであるべきである。クライアントサイトでは、訂正は手動でなされてもよいし、あるいはそのように構成されている時には、クライアントが転送に参加できるようにリソースを解放するサーバからの遠隔制御の下になされてもよい。

全送信期間のどの時点においても、クライアントはこのパケットに応答して、そのエンドからの送信をアボートしたことを示し、メッセージ中にその理由を示すことができる。もし転送が完了以前に中断されたのなら、本発明では、ファイル中の既に送信に成功した部分を再び送ることなく、後に続行する（図1の「再開」）ことができる。このことは、非常に大きなファイルを送るときには、特に重要で有用な特徴である。この特徴を実現するために、クライアントは、部分的に受信済みのファイルを破棄しない。その代わり、クライアントは、部分的に受信済みのファイルを格納する。もし、ファイルが最初に送られている時、すべてのクライアント（例えば、１マルチキャストグループ内のすべてのクライアント）がそのファイルの全体を受信できないようにする問題が発生（例えば、ファイル送信中に何らかの理由でリンクが切れたときなど）すれば、後に送信は再開され、転送を完了することができる。再開時には、サーバは、受信されなかったデータフレームのリストをすべてのクライアントにクエリー（query）した後、サーバは、それらのフレームのみを送ることによって、転送を完了させ始める。よって、図1では、再開にあたって、ステップ10は、転送がアボートされなかった場合の通常の開始時のように、ファイルの最初のブロックの最初のフレームから開始するのではなく、初回のアボートされた送信の間に受信されなかった（つまり、否定応答確認された）フレームからまず開始する送信を伴う。

転送

データ転送フェーズに入ると、送信ログは、サーバにおいて保守される。このログは常にオンであり、すべてのイベントを見失わないようにする。それぞれのクライアントもまた、送信ログを保守する。それぞれのクライアントで保守されるログは、後に「完了」の項で参照する。

2ギガバイト以上のデータを有するファイルが転送されうる時、転送の全期間にわたってファイルの全体をサーバのメモリに保持しておくことは、一般に現実的ではない。そのファイルを受信することになるクライアントの数は、1000以上でありうるので、次のブロック転送に引き続いて移る前に、送信を一時中断して、それらのクライアントのそれぞれからの応答確認を待つことは、受け入れがたいことである。

サーバは、転送されるそれぞれのファイルを、フレームのブロックへと論理的に分解する

。それぞれのブロックは、典型的には、複数のフレームを有しており、何千ものフレームを有する可能性もある。図4を参照すると、ある例では、サーバ20は、１個のファイルを４つのブロック、すなわちブロック1、ブロック2、ブロック3およびブロック4に分解している。ここで、それぞれのブロックは、１つ以上のフレームを有している。それぞれのブロックは、１個のブロックがサーバにより送られたとクライアントが判定したとき、転送に参加しているあらゆるクライアントにより否定応答確認される（だけで、肯定応答確認はされない）単位を表す。クライアントは、このことを、受信されたデータパケット中のブロック番号の変化により検出する。なぜなら、送られたそれぞれのフレームは、そのブロック番号と、そのブロック内でのフレーム番号とを示すからである。ファイルを複数のブロックに分解することにより、少なくとも２つの利点が得られる。すなわち、（ｉ）要求される否定応答確認の個数を減らすことができ、かつ（ｉｉ）次のファイルパスの転送ブロックを決めるためのサーバでのメモリへの要求を減らすことができる。

データの転送は、否定応答確認に直接、結びつくものではない。転送は、どのような個別のクライアントによる否定応答確認を受信しそこねたとしても、あるいは、どのクライアントが、以前にデータパケットが受信しそこねたとしても、継続する。これによって、設計を簡単にすることができ、個々のクライアントの問題が、グループ全体には最小限の影響しか与えないようにすることが確実になる。なお、クライアントは、サーバから受信した内容に基づいてブロック否定応答確認を送ることについては、責任があることにも留意されたい。

図4を参照すると、サーバは、まず、第１のブロックの第１のフレーム（つまり、ブロック$_1$の第１のフレーム）を送ることによって、転送を開始する。サーバは、これらのフレームを設定可能なレートで送る。このことは、基本転送レートが、パフォーマンス次第で減速させる（つまり、低くする）ことができることを表している。サーバは、完全なファイルがいったんネットワークに送られてしまう（つまり、ブロック$_1$～ブロック$_4$が送信される）まで、ファイルのフレームを送り続ける。これが、第１パスまたは第１ラウンドとして規定され、図4において「B_4」として表されているだけの量の時間を要する。クライアントの中には、第１パスの後、完全なファイル（つまり、４つのブロックのすべて）を正しく受信できた者もいる。そのような場合、それらのクライアントは、ファイルの受信を終了したことになる。１個以上のエラーを含むフレームを受信したか、または１個以上のフレームを全く受信していないクライアントは、そのファイルのいくつかの「部分」（つまり、誤って受信したフレーム、または受信しそこねたフレーム）を、後続するパスまたはラウンドで再び送ることを要求する。後続するパスまたはラウンドはそれぞれ、前回よりも少ない数のフレームの送信を要求する。なぜなら、前回のラウンドで否定応答確認されたフレーム（つまり、受信されなかったか、あるいは誤って受信されたフレーム）が、後続するラウンドでは再送信されるからである。

完了するための最大パスカウントまたは最大時間は、設定可能なパラメータでありうる。最大パス時間つまり最大持続時間までに、ファイルの全体を正しく受信できたわけではない複数のクライアントが存在する可能性がある。これらのクライアントはサーバにより識別される。また、サーバは、例えば、ユニキャストファイル転送プロセスを介して、残りの情報をこれらのクライアントが得ることができるようにするために、さらにアクションを起こすことができる。好ましい実施の形態では、クライアントは、「ダン」メッセージを送ることによって、ファイル全体を受信したことを示し、一方、サーバは、「ダンリスト」を送ることによって、「ダン」であるといっているクライアントを示す。もし、ある所定のイベント（例えば、所定の時間経過）の後、サーバが、いくつかのクライアントから「ダンメッセージ」を受け取っておらず、すべてのNAKがサービスされているのなら、サーバはそれらのクライアントに対してステータスリクエストメッセージを送り、さらなるデータを必要としているクライアントには、欠けているすべてのフレームを送る。依然として応答しないどのようなクライアントに対しても、例えばユニキャスト転送により、その後、サーバからそのクライアントへとファイルを送ることができる。

サーバがブロック境界（すなわち、図4ではB_1、B_2、B_3およびB_4）を通る時、個々のクラ

イアントは、好ましくは、それぞれのブロックについて「マルチプルセレクティブリジェクト否定」応答確認（「Nak」）を送る。それぞれのブロックについてクライアントから送られてくるこれらの応答確認は、そのブロックの境界通過後、しばらくしてからサーバにおいて受信される。肯定応答確認は、暗黙のうちに示される。ある特定のブロックに対するマルチプルセレクティブリジェクト否定応答確認は、その特定のブロックにおける１個または多数のフレームが、これらのクライアントにより誤って受信されたか、あるいは、全く受信されなかったことを意味しており、何らかの理由でネットワークがそれらのフレームを伝達しなかったことを示している。よって、サーバに送られた応答確認は、誤って受信されたり、全く受信されなかったフレームがどれであるかを示している。

後続するパス（すなわち、図4に示す第１パスに続くパス）では、クライアントは、再び正しく受信されなかったブロックについての否定応答確認のみで応答する。サーバは、後続するパスで、さまざまなクライアントにより要求されたファイルの部分（フレーム）をすべてのクライアントに送るので、クライアントの多くが、それを既に第１パスで正しく受信している可能性がある。よって、それらのクライアントは、それを無視することになる。

一般に、クライアントからサーバに戻ってくるすべての情報は、サーバがそれらのフレームをクライアントに転送するのに用いる（１つ以上の）パスとは別の戻りパス上で送信されうる。しかし、ここでは、説明のために、通信リンク24（図2）または、サーバとクライアントとが通信できるようにするその他のパスは、サーバからクライアントへのリンクおよびクライアントへの戻りリンクの両方を広く意味するもの解釈されたい。

サーバは、転送およびその転送の加入者に関するさまざまな情報を保守する。好ましい実施の形態では、この情報は、データ構造またはリストのかたちでサーバにより保守される。サーバは、この情報を保守し、使用することによって、ファイル転送のステータスを記録し、決定する。

また、サーバは、すべてのクライアントからの個々のフレームに関するセレクティブリジェクトのすべてを示すフレームデータ構造も保守する。もし多数のクライアントが同一のフレームを受信しそこなったのなら、フレームデータ構造は、そのフレームが受信されなかったことを示すのみである。すなわち、フレームデータ構造は、サーバによりクライアント単位で保守されるわけではない。一般に、サーバが、受信されなかったフレームの詳細なリストを個々のクライアント単位で保守することは望ましくない。なぜなら、そのようなスキームは、特に多数（例えば、1000以上）のクライアントがマルチキャストに参加している時には、法外な量のメモリを用いることになるからである。例えば、１以上のクライアントが、ブロック$_1$のフレーム24および25、ブロック$_2$のフレーム1、ブロック$_9$のいくつかのフレーム…を全く受信しなかったか、あるいは誤って受信したものとする。もしサーバにより保守されているフレームステータスが、ある特定のブロックのある特定のフレームが再送信される必要があることを示すのなら、クライアント中の少なくとも１人が、その特定のブロックを首尾よく完了したことを応答確認していないことは事実である。サーバがファイルの全体をいったん送信した後、サーバは、このフレームステータス情報を通過させ、そこにリストアップされているフレームのみを再び送信する。この操作は、すべてのクライアントが「ダンメッセージ」を送り終わり、フレームステータスリストが空になる（つまり、最大ラウンド数あるいは最大時間に到達する）まで、１パス毎に継続する。

なお、与えられたどのパスについても、もし否定応答確認が全くサーバに戻ってこないのなら、クライアントは、サーバによる次のパスの間、同じリジェクトおよび再送信リクエストメッセージを送り返すことになることには注意が必要である。このことは、もしある特定のクライアントがサーバにより受信されていないのなら、そのクライアントは、より長い時間のあいだ参加しなければならないが、そのクライアントが、残りの受信クライアントに顕著な影響を及ぼすことはないことを意味している。

サーバに格納されている情報の別の一部としては、マルチキャストグループに関する統計がある。送信が完了すると、オペレータがシステムパフォーマンスの問題および／または

特定のクライアントのパフォーマンスの問題を判定する一助になるサマリー情報が送信データにのせられる。

ファイルを通した多数のパス：

いったんファイルが一回、完全に処理されると（つまり、第１パスまたは第１ラウンドが終了すると）、本発明による送信プロセスは、パスカウンタをインクリメントし、エラーのあった最初のブロックについて、サーバにあるフレームステータスリストをスキャンする。この最初のエラーブロックを発見するとただちに、サーバは、そのブロック中の受信されなかったパケットを再び送る。これらの受信されなかったパケットに対する否定応答確認は、前述したように、クライアントがブロック内のエラーを検出した時に、クライアントにより発生される。これは、第１パスとも一貫している。セレクティブリジェクト否定応答確認はすべて状態を示すものであるので、あるパスに固有ではない。ただし、それらの否定応答確認は、それぞれのパスで変化することはある。好ましい実施の形態では、マルチプルセレクティブリジェクト否定応答確認は、１個のワードの全体が１個のブロックを表し、そのワード中のそれぞれのビットが、そのブロックを構成する複数のフレームのそれぞれ異なる１つを表す、ビットマップのかたちをとる。

送信アボート：

もし、送信中に、修正不可能な欠陥に遭遇したか、あるいは、オペレータが手動でアボートしたのなら、送信アボートシーケンスが開始される。このシーケンスは、ある間隔のあいだ（例えば、送信ファイルにおいて指定された間隔のあいだ）、アボートメッセージを繰り返し送信することを必然的に伴う。レシーバは、このアボートメッセージを応答確認し、例えば、転送の潜在的続行（つまり再開）のためのコンテクストを保存するか、または、別の送信に備えてコンテクストを再び初期化するアクションを起こすことができる。ユーザが、送信アボートを初期化することを可能にする設備がある。理由コードは、一時中断のためにも、あるいは初期化のためにも設定されうる。前者の場合、送信は、その後ある時刻に続行または再開されうる。一方、後者の場合、クライアントは、そのコンテクストを再び初期化するようにリクエストされる。

完了

サーバは、クライアントから「ダンメッセージ」を受信することによって、個々のクライアントの完了を検出する。クライアントは、自分にファイルのすべてのブロックが手に入るとすぐに終了したことが分かるが、サーバが完了を確認するまでは、「ダンメッセージ」を送り続けなければならない。サーバは、あるクライアントが「ダン」であることを、そのクライアントのアドレスを「ダンリスト」に入れ、そのリストを複数のクライアントに送り出すことによって、確認する。クライアントは、自分のアドレスが「ダンリスト」にリストアップされているのを見て、転送を完了したことを知る。その後、このクライアントは、その送信ログを更新することによって、転送が上首尾に完了したことを示す。

サーバまたはクライアントからの転送をアボートする能力も設けられる。アボートパケットは、サーバおよびクライアントに対して、転送を早くにアボートする能力を与える。もしクライアントがアボートを送れば、サーバは、そのクライアントをグループから除く。もしサーバがアボートすれば、転送は、ファイルの全体を第１パスで送ってしまわなくても、再開可能である。

ステータスリクエスト：

もし、第１パスの後、サーバがクライアントからDONEもNAKも受信しなかったのなら、ステータスが分かっていないクライアントに対してクエリーが直接、送られる。これらの応答は、標準的な応答メッセージのかたちをとる。また、これらの応答は、もしリポートすべきエラーがあるのなら、それらのエラーを記述するビットマップを含むことがある。

輻輳／フロー制御

大型の「インターネット」がマルチキャスト可能になるにつれて、情報が、グループのメンバーたちに対してそれぞれ異なる送信リンクをもつことを望むマルチキャストグループは、さらによく見られるようになるであろう。これらの異なるリンクは、それぞれ異なる容量を有することができる。これら容量は、互いに大きく分散していてもよい。例えば、

グループ中のあるメンバーが1Mbpsを超えるリンク容量をもつ一方で、別のメンバーが、わずか56Kbpsをもっていてもよい。一般に、これらのリンク容量に関する知識は、送信波の送り手（例えば、サーバ）には分からないであろう。よって、リンク容量を瞬時に決定し、フロー制御メカニズムを設けることによって、ネットワークのオーバーロード／輻輳を防止するとともに、データ転送プロトコルの効率を制御しないようにできるのが望ましい。

ここで説明されているデータ転送プロトコルは、ブロックの概念を含んでいる。これらのブロックはそれぞれ、何百または何千ものフレームを含みうる。クライアント（受け手）は、もしそのブロックにおいて何らかのフレームが欠けているか、あるいは誤っているのなら、ブロックの境界において、マルチプルセレクティブリジェクトNAKを送ることを強制される。フロー制御の目的から、実用的である限りにおいてできるだけ早く、受信されなかった／誤りを含む（つまり、脱落している）フレームに関する知識を得て、フロー制御の判定を下すことができるようにするのが望ましい。本発明によるデータ転送プロトコルを用いてこれを達成するためには、変更可能な、つまり可変のブロックサイズを使うのが、一つの方法である。このことは、比較的小さなブロックから始めて、ファイル転送の間にブロックのサイズを大きくしていき、クライアントの応答確認を減らすことによって、現在のスケーラビリティを維持することを必然的に伴う。本発明によるデータ転送プロトコルを用いてこれを達成するための別の好ましい方法としては、ブロックのサイズはすべて同じに（均一なブロックサイズに）するが、ブロック境界よりも前にクライアントがNAKで応答することができるように、ブロック境界が発生する前に、サーバにステータスリクエストを遅らせる方法がある。この後者の技術のほうがフレクシブルである。なぜなら、NAKは、ブロック境界のみ（前者の技術はこのケースに相当する）ではなく、いかなる時にも請求される可能性があるからである。これら２つの技術のいずれによっても、NAKは、転送の早い段階で請求される。

「可変ブロックサイズ」方法（前段落で言及した第１の技術）では、第１のブロックは、比較的小さくてもよい（例えば、100フレーム）。後続ブロックは、毎回２倍ずつ大きくなっていく。ブロックサイズは、最大のブロックサイズに到達するか、またはファイルがその終わりに達するまで、次々に倍増されていく。

「ステータスリクエスト」方法（前段落で言及した第２の技術）では、サーバは、自らが望む時点でNAKリクエストを請求する。それらの時点は、ブロック境界ではない。輻輳またはフロー制御のための好ましい実施の形態では、ステータスリクエストは、回を追うごとに長くなる間隔で送られる。

両方の方法ともに、伝送レートまたは転送レートは、本明細書に記載されているように設定される。しかし、固定された転送レートではない、セット可能な転送レートは、転送レートの上限を表す。第１のブロックの後（可変ブロックサイズ方法の場合）またはステータスリクエストが受信された後（ステータスリクエスト方法の場合）、NAKが、脱落したフレームのあるクライアントによりサーバに送られる。また、これは、これらのクライアントによりサーバに輻輳が示されることでもある。

もしNAKがあるのなら、それらが特定のリンクの瞬時の容量に関連しているという事実は、以下の方程式に基づき、輻輳を示すリンクのすべてについてリンク容量を決定するのに用いることができる。

（（送られたフレーム数－否定応答確認されたフレーム数））×転送レート＝リンク容量

図6の異質マルチキャストネットワークでは、リンク速度は、64Kbps～1024Kbpsの範囲であり、リンク容量には大きな差がある。その他のトラヒックがないものとすると、もし転送レートが150Kbpsに合わせて設定されれば、クライアントAからのブロック1に対するブロックNAKは、（可変ブロックサイズ方法では）第１のブロックについて約58個のフレームが脱落していることを示すことになる。上記方程式を用いれば、瞬時のリンク速度は、63Kbpsと計算される。クライアントBからのブロック1に対するブロックNAKは、（可変ブロックサイズ方法では）第１のブロックについて約15個のフレームが脱落していることを示すことになる。再びこの方程式を用いれば、Bへのリンクの瞬時のリンク速度は、127.5

Kbpsと計算される。その他のトラヒックが存在する場合、脱落するフレームの個数はより多くなるので、計算されるリンク速度はさらに低くなる結果となる。
グループ閾値パラメータは、ユーザにより設定されうる。グループ閾値は、マルチキャストグループに加入し続けることが許可されている特定のクライアントによる限界（脱落しているフレームの占めるパーセントで表される）である。もしグループ閾値が25%に設定されれば、グループ内で25%よりも高いフレーム脱落パーセンテージをもつどのクライアントも、グループの残りの者が悪影響を被らないように、アクションを起こす必要があることを意味している。図6の例では、58%のフレームが脱落しているクライアントAは、アクションを起こす必要があることになる。クライアントは、判定を下すのに十分な情報をもつことになる。なぜなら、転送レートおよびグループ閾値パラメータは、アナウンスメッセージのかたちでクライアントに送信されるからである。自分のフレーム脱落レートが閾値を超えていることを検出したクライアントは、以下のアクション中の一つを起こす。
1. グループを去り、より低速のグループに入ることをサーバからリクエストする。ここで、そのグループは、そのクライアントにおいてなされた測定に基づいて速度指定される。
2. それ以上の伝達をリクエストせずにグループを去る。このことは、このクライアントがこの送信を受信しそこなうことを意味する。
3. ステータスリクエストメッセージがサーバから受信されるまで、NAKを抑える。これにより、フレームロスの大きいクライアントから過度の再送信により停滞させることなく、グループの残りの者が終了できるようになる（このクライアントのセットに対する再送信のための転送レートは、それらの容量が小さいことを反映するように低くすることができる）。
図6の例では、２番目に高いフレーム脱落パーセンテージは、クライアントBの15%である。この値は、グループ閾値よりも低い。この数は、パフォーマンスを過度に低下させることなく、グループ全体が合わせることのできる係数を表している。クライアントBに合わせるためには、このグループのサーバ転送レートは、15%、またはそれよりも高いパーセンテージ、またはそれよりもわずかに高いパーセンテージだけ下落することになる。
可変ブロックサイズ方法のタイミングは、図8に示されている。クライアントにおける情報が、そのフレーム脱落はグループ閾値を超えていることを示すと直ちに、そのクライアントは、グループの送信に悪影響が及ぼされないように、上に列挙した３つの選択可能なアクションの中から１つを選択しなければならない。グループの転送レートの調整は、第２のブロックが送られた後おこなわれ、ブロック3のはじめから開始する。転送レートの変化は、ブロックの境界で実施され、正確なデータをブロック単位でブロックNAKから供給する。その後、ファイル転送は、ブロック3の転送に進む。ブロック3は、ちょうどブロック2がブロック1の２倍の大きさであるように、ブロック2の２倍の大きさに設定される。この後、ブロック4が続く。ブロック4は、ブロック3の２倍の大きさである。以下も、最大のブロックサイズに到達するか、ファイルがその終わりに到達するまで（それらのどちらが最初に起こっても）同様である。しかし、もしブロック3以降のグループからのNAKが、最悪のクライアントでもレート閾値パラメータ（設定可能）を超えていることを示すのなら、そのレートは、ブロック5の送信のためにさらに調整される。このレート閾値は、グループに対する転送レート調整がおこなわれる最小のフレーム脱落パーセンテージである。例えば、クライアントからの最大フレーム脱落パーセンテージが1%であるのなら、レート閾値が典型的には、1%を超える数に設定されるような調整は保証されない。
ステータスリクエスト方法では、ブロックは均一なサイズであり、ステータスリクエストは、ブロックの境界に到達する以前に、サーバにより送られてNAKをリクエストする。図9を参照すると、可変サイズブロック方法について今説明したばかりのシナリオと等価なシナリオが図示されている。ただし、ここでは（図9）では、ブロックサイズは均一である。ある例では、第１のステータスリクエストは、100個のフレームを転送した後に送られ、第２のステータスリクエストは、200個のフレームが送られた後に送られる。以下も同様である。クライアントのNAKは、可変ブロックサイズ方法と正確に同じ時刻にサーバに

送り返される。しかし、ステータスリクエスト方法では、可変ブロックサイズ方法のようにNAKを受信するためにブロック境界を待つ必要はなく、いつでも望みの時刻にステータスリクエストが送られるという点で、よりフレクシブルである。
可変ブロックサイズ方法でも、ステータスリクエスト方法でも、単にグループのメンバーを削除し、それらのメンバーをハンギングのまま放置することは一般に望ましいことではない。削除されたグループメンバーは、より低い転送レートで動作する別のグループに集めることができる。より遅いこの転送レートは、グループを去ったクライアントによりおこなわれるリンク容量の計算により決定されうる。その後、このグループは、ふさわしい転送レートでセットアップされうる。そして、新しい転送が開始されうる。
フロー制御プロセスの可変ブロックサイズ方法およびステータスリクエスト方法はともに、自動化されうる。
マルチキャスト：
マルチキャストは二つの形をとりうる。すなわち、ネットワークが依然としてデータをブロードキャストグループの全体に伝達しているアプリケーション層（AL）マルチキャストと、ネットワークが、マルチキャストルータに基づいてトラヒックをルーティングしており、インターネット仕様RFC1112がクライアントに実施される、マルチキャストIPの２つである。
いずれの場合でも、マルチキャストグループは、サーバの開始の下にセットアップされる。サーバは、ユニキャスト単位でクライアントに通知を送ることによって、クライアントに対して特定のマルチキャストグループのメンバーシップを通知する。これらのマルチキャストグループは、迅速にセットアップされ、解体されうるので、マルチキャストグループをダイナミックに構成することが可能となる。例えば、マルチキャストグループは、ある特定のファイルを送信するためだけにセットアップされうる。その後、このグループは解体されうる。
ALマルチキャストでは、ネットワークは依然としてブロードキャスト単位でトラヒックを伝達しているが、グループに属していないクライアントは、それ専用ではないデータを破棄する。グループがセットアップされると、たとえデータがそのノードで破棄されないような事態が発生しても、グループの外のクライアントがそのデータをリードすることができないように、セキュリティキーを領布することもできる（なお、このことは、マルチキャストIPでも展開されうる）。また、ALマルチキャストでは、IPアドレスは、グローバルな、またはネットワークベースのブロードキャストアドレスのままである。ブロードキャストの場合と同様に、このアドレスは、リンク層プロトコルにおけるブロードキャストアドレス（例えば、ブロードキャストSMDSアドレス）にマッピングされる。マルチキャストヘッダが、このグループについて選択され、グループの識別子となる。
マルチキャストIPの場合、ネットワークは、ルータがクラスDのマルチキャストIPアドレスおよびマルチキャストルーティングをサポートする、ルータネットワークである。クライアントは、RFC1112、すなわち「IPマルチキャスティングのためのホスト拡張」をサポートする。RFC1112は、ルータテーブルのアップデートを目的として、最も近いマルチキャストルータへとその存在をホスト通知することを可能にする。
以下に、上述した本発明の機能について説明する。
再び図2を参照する。図2は、あらゆるブロードキャストまたはマルチキャストIPルータベースのネットワークを広く表しうる。本発明の目的は、ワイドエリアネットワーク（WAN）接続24を介して、サーバ20により5000以上の受信ノード22へと、小型または大型のデータファイル（サイズが2ギガバイトまで、またはそれ以上のファイル）を同時に送信可能とすることである。また、本発明は、前述したように、ローカルエリアネットワークまたはその他のタイプの通信リンク上でも作用可能である。送信媒体24は、好ましい実施の形態においてTCP/IPプロトコルスタックをサポートできれば、どのようなタイプのものでもよい。その他のプロトコルスタックもまた、本発明のための通信環境として作用可能であろう。
マルチキャストは、２つの方法によりサポートされうる。すなわち、前述したように、AL

マルチキャストと、マルチキャストIPとの２つである。
クライアントに転送されるファイルは、テープを介して（例えば、図5のテープドライブ66を介して）サーバ20にロードされうるし、もしそれらのファイルが十分に小さければ、フロッピー（例えば、図5のフロッピードライブ）によってもロードされうる。また、転送されるファイルは、例えば、LANあるいはその他のネットワーク上で、ファイルのソースから、FTP（ファイル転送プロトコル）あるいはその他のユニキャスト転送メカニズムを介して、サーバ20へとロードされうる。これらのファイルは、一般に、どのようなフォーマットであってもよい。その後、データファイルが、テープまたはフロッピーから、送信サーバ20のファイルシステム内へとリードされる。なお、サーバ20が、データファイルの圧縮されていないコピーをリードするのに十分なスペースをもっていなければならないことには、留意されたい。どちらのサービスについても、データファイルは、不適格なレシーバがこのデータファイルを受信して利用することができないように、暗号化されてもよい。送信ファイルはそれぞれ、好ましくは、ユニークに識別される。その内容および発生時刻については、好ましくは、指示がある。プロセスへの入力ファイルは、2ギガバイトを超えるサイズであってもよい。また、システムは、2ギガバイトよりもはるかに大きいファイルを操作することもできる。
その後、ファイルは、サーバ20に格納され、送信の準備が整えられる。以前の送信からのデータは、再送信されることが必要である場合には、しばらくのあいだサーバ20上でいつでも利用できるようにされる必要がある。このデータにアクセスするためのメカニズムは、そのデータを待ち行列に入れておいて、いつでも再送信できるように設けられる。
効率のために、ファイルはブロックのかたちで送信される。ブロックのサイズは、通信リンク24上で転送されうる最大のパケットから導き出される（あるいは、ブロックサイズは、ユーザにより選択されてもよい）。その導出は、クライアントが、１個のブロック中のどのパケットを受信しそこねたかをサーバに示すことが必要になるという事実に基づいている。これをおこなうためのある方法（そして、一般に最も簡単な方法）としては、ビットマップを送って、ビットの設定値により、どのパケットが受信されなかったかを位置的に示す方法がある。よって、ブロックのサイズは、それ自身は１個のパケットの中に含められるビットマップのかたちで応答確認されうるパケットの個数にほぼ等しい。例えば、もしパケットサイズが256バイトであれば、１個のパケットが含むことのできる最大のビット数は、256（バイト／パケット）×8（ビット／バイト）＝2048（ビット／パケット）となる。このことは、許容される最大のブロックサイズが、2048個のパケットを有するブロックであるということを意味する。
受信ノード22を10MbpsでイーサネットLANにインタフェースさせることは可能ではあるが、WANリンクは、それよりもずっと低い速度であることが多い。よって、明示的な送信データレートは、セット可能／設定可能である。
受信ノードは、それぞれ、送信の前、または送信中にリソースの問題を経験する可能性がある。受信ノードは、送信前にそのリソースについてクエリーし、データを受信するための設備をもっているかどうかを判定するようにイネーブルされる。もしそうではないのなら、これらのノードは、送信専用のスペースを再び初期化すべきであるか、あるいは、送信に参加できないことを示すべきである。そうすれば、異なるチャネルを通して、修正対策に着手可能となる。サーバが遠隔操作により利用可能なディスクのスペースを強制的に作らせることによって、ファイルの転送を実行可能とする設備を設けてもよい。
レシーバ22は、現在何を聴取しているかに気づいていなければならない。ある専用チャネルでデータグラムが受信される時、ノード22は、自らがアドレシングされているかどうかを判定しなければならない。このアプリケーションが２以上の送信サーバ20により用いられている時には、問題が発生する。受信ノード22が、ある与えられた時刻に正確に一つの送信に参加していることを保証できる方法がなければならない。UDPポートをサーバ20に専用とし、かつ暗号鍵をそのサーバに関連づけることによって、ネットワーク24上でランダムなモードタップを用いている受信ノードが、送信されたデータを解釈できる能力をもつような事態を確実に回避できる。

送信サーバ20上では、何らかの参照情報が保守されている。好ましくは、ネットワーク内のすべての可能な受信ノードのリストがある。サービス故障、問題などの場合に、情報プロバイダがクライアントを管理することを可能にするのに十分な参照情報が利用可能であれば好ましい。また、暗号化され、圧縮されたデータファイルがいつでも送信できるように保守されている送信データベースがあれば好ましい。この送信データベースは、準備されたデータとともに、ファイルの内容を識別する、例えば、70バイト以下の記述情報を含んでいる。

それぞれの送信は、好ましくは、完了ステータス指標記録と、送信中に遭遇したすべてのエラーのログとを有している。また、送信に失敗した対象であり、データを送るべき全ノード、および失敗した理由を記したリストを含むイベントファイルがあれば好ましい。

送信中のどの時点においても、オペレータは、サーバ20およびそれぞれの受信ノード22に適用している送信のステータスを問い合わせることもできる。もし、いくつかのクライアントへの通信に関する問題、あるいはその他の問題が発生すれば、警告が発せられる。もし何らかの介入が示されれば、オペレータには、修正アクションを開始することが許可される。

サービスに関する現在進行中の保守および管理について、オペレータには、レシーバ、送信グループ、送信ファイル記述子、送信パラメータおよび送信データベースのリストを保守することが許可される。バックグラウンドプロセスは、環境および年齢データの両方を保守し、もし警告を受けたオペレータにより許可されれば、ハウスキーピングパラメータにより、それを削除することになる。

以上に、本発明によるデータ送信について説明した。以下では、本発明のその他の局面について説明する。その他の局面には、「セット可能な伝送レート」、「マルチキャストグループ」、「マルチキャストピン」、「マルチキャストネットワークプローブ」、「速度グループ」および「否定応答確認収集」がある。

セット可能な伝送レート

前述したように、データ伝送レートをセットすることが可能である。以前に述べた例では、セット可能なレートが役に立つ場合を説明した。その例では、受信ノード22は、利用可能な帯域幅が10MbpsであるイーサネットLANと、そのLANを10Mbpsよりもはるかに低い速度のその他のネットワークに接続するWANリンクとにインタフェースされる。このような場合、データ伝送レートは、本発明によれば、例えば、最も低速のWANリンクの速度と一致するようにセットされることになる。

本発明によれば、どのようなファイル転送セッションが与えられたとしても、データ伝送レートは、事前にセットできる。もっと詳しく言えば、そのセッションの間にデータが送信される際の最大ビットレートは、セット可能である。好ましい実施の形態では、最大ビットレートは、キロビット／秒（Kbps）単位でビットレートを表す整数値にパラメータを設定することによってセットされる。例えば、もしこのレートパラメータが値56であるのなら、このパラメータは、最大のビットレートである56Kbpsに対応する。このレートパラメータは、ソースをデスティネーションに接続するリンクの利用可能な帯域幅を対応するいかなる値、または利用可能な帯域幅未満のレートを表す値に設定されうる。すなわち、もし利用可能な帯域幅が1Mbpsであるのなら、レートパラメータは、0から1000の間のどの値に設定されてもよい。ここで、1000Kbpsは、1Mbpsに等しい。転送レートを明示的にセットする能力は、ネットワーク全帯域幅あるいは実質的にすべての帯域幅を占有することなく、ネットワーク上のその他のアプリケーションと、（時間的に）長いファイル転送とを共存させることを可能にする。

マルチキャストグループ

「マルチキャスト」は、以上の説明では、サーバノード20が、ネットワーク24に接続されているすべてのクライアントノード22の１サブセットにデータ（例えば、ファイル）を送る場合として規定された。また、以上の説明では、マルチキャスト送信は、２つのかたちをとりうることも開示された。すなわち、「アプリケーション層（AL）マルチキャスト」と、「マルチキャストIP」の２つである。ALマルチキャストは、ネットワークがインター

ネット仕様RFC1112はサポートしないが、ブロードキャストはサポートする時に用いられる。もしマルチキャストIPがRFC1112およびマルチキャストIPルーティングに従ってネットワークによりサポートされるのなら、マルチキャストIPのほうが、ALマルチキャストよりも勧められる。マルチキャストIPは、グループのメンバーがマルチキャストをサポートしなければならず、ルータネットワークにおけるルータもまた、ある種のマルチキャストルーティングプロトコル（例えば、DVMRP、MOSPFあるいはPIM）をサポートしなければならない時に用いられる。ALマルチキャストとは異なり、マルチキャストIPは、マルチキャストグループのメンバーのみが送信されたデータを受信する、真のマルチキャストプロトコルである。

それぞれのファイル転送のたびに、マルチキャストグループは、前述したように、データ送信の「アナウンス／登録」局面のあいだに規定されうる。既に述べたように、サーバは、ファイル転送およびその転送に参加している加入者またはグループに関するさまざまな情報を保守する。好ましい実施の形態では、この情報は、データ構造またはリストのかたちでサーバにより保守される。サーバは、この情報を保守し、用いることによって、「データ転送」段階のあいだのファイル転送のステータスを記録し、決定する。クライアントステータス構造は、サーバにより受信されたアナウンス登録からのデータに基づく、マルチキャストグループの加入者のステータスリストを含んでいる。

マルチキャストグループ管理は、クライアントをマルチキャストグループに割り当てるプロセスである。それぞれのグループのクライアントリストを作成し、操作するタスクは、最初の事例でファイル転送を開始するアプリケーションプログラムの責任である。アプリケーションプログラムは、一般に、名前をクライアントのIPアドレスに関連づけること、名前をグループに割り当てることのような、簡単に利用できる特徴を提供する。グループ管理は、送信側の局（例えば、サーバ）のみで要求される。マルチキャストグループは、送信側の局がファイルを送信したいと望む時に指定される。このグループは、クライアントのIPアドレスリストにより識別される。ここで、１個のアドレスが、マルチキャストグループ内のそれぞれのクライアントに相当する。

マルチキャストグループには２つの選択肢がある。すなわち、ダイナミックおよびスタティックの２つである。ダイナミックマルチキャストグループの場合、転送が完了すると、グループは解体する。ダイナミックマルチキャストグループは、マルチキャストグループのクラスDアドレスを用いる「アナウンス」メッセージにより形成される。ダイナミックマルチキャストグループとは対照的に、スタティックマルチキャストグループでは、転送が完了しても、グループのメンバー全員がそのグループのメンバーのままである。スタティックマルチキャストグループは、サーバによりユニキャスト単位で形成され、および／またはコンフィギュレーションをセットアップするために通常のクラスDアドレスを用いて形成される。

マルチキャストピン

TCP/IPにおける「ピン」ユーティリティは、TCP/IPネットワーク内の２つのポイント間のコネクティビティを判定する（すなわち、２つのポインタが実際に接続されているかどうかを判定する）際には非常に有用である。TCP/IPでは、ピンパケットが所望のエンドポイントに送られると、そのエンドポイントは、アドレスを反転させ、それを送り手に送り返す。また、周回時間遅延も測定される。これは、ピンパケットが送り手から所望のエンドポイントへと移動してから、送り手に戻ってくるまでの時間の測定である。

また、あるマルチキャストグループのメンバー全員がピンパケットまたはピンリクエストに応答する、マルチキャストピンユーティリティを提供するのも望ましい。マルチキャストIP（RFC1112）をサポートするクライアントまたはホストは、ピンリクエストに対して、デスティネーションアドレスとしてのクラスDのIPアドレスで応答する。しかし、現在知られているマルチキャスト実現形態では、ピンリクエストの送り手は、そのピンリクエストに対して受信した最初の応答しか表示しない。すなわち、現在知られているマルチキャストピン技術では、ネットワークコネクティビティの測定をおこなわない。

本発明による「マルチキャストピン」という特徴によれば、ピンリクエストに対するすべ

てのマルチキャスト応答を表示することによって、ネットワークコネクティビティ情報をソースからグループ内の受け手へと提供し、マルチキャストグループ内のそれぞれの受け手についての周回時間遅延情報を提供する。好ましい実施の形態では、この特徴は、標準的なピンICMPメッセージを用いる。

また、本発明による一改善事項として、前述したアナウンス／登録機能を、「マルチキャストピン」という特徴の別の形態として用いることも可能である。この改善をおこなうことにより、アナウンス／登録ピンメッセージは、グループ内の受け手のアプリケーション層に至るコネクティビティおよび送り手に戻る側のコネクティビティを判定し、グループ内のそれぞれの受け手について周回時間遅延情報を判定する。

よって、この「マルチキャストピン」という特徴により、マルチキャストグループのメンバーについて、送り手がネットワークコネクティビティおよび周回遅延を判定することが可能になる。

マルチキャストネットワークプローブ

マルチキャスト（１つのものを全員ではなく、多数に送ること）データネットワークは、今まさにその実現段階に移行しようとしている。特に、マルチキャストIPは、ルータネットワークの中でも新しく、あらゆる種類（例えば、フレームリレー、SMDS、LAN、衛星、無線など）のネットワーク上でマルチキャストグループをつくるためのメカニズムを提供できる。インターネットもまた、「Mbone」（マルチキャストバックボーン）、すなわちマルチキャストIPをサポートするインターネットの一部を有している。

Mboneは、1992年の早いうちから開始され、成長を続けた結果、1995年のはじめには、インターネット中の1500を超えるサブネットがマルチキャスト可能になった。今日に至るまで、Mboneは、インターネット「ラジオ」およびその他の実験的アプリケーションとともに、PCおよびワークステーションに基づくビデオ会議およびホワイトボードマルチキャストアプリケーションをテストしてきたインターネットの研究者たちにより、実験的ネットワークとして用いられてきた。Mbone上のマルチキャストIPルーティングは、当初、マルチキャストルーティングプロトコルDVMRPを用いてワークステーションにおいて実施された。しかし、Mboneの一部は既にそのルータをアップグレードしていたので、それらは、現在、マルチキャスト可能である。今後、５、６年のうちに、インターネットは、インターネット内のルータを用いることによって、完全にマルチキャスト可能になると予想されている。

インターネット中のマルチキャスト可能な部分が増えていくにつれて、Mboneは、単なる実験用の研究ツールとしてではなく、メインストリームマルチキャストアプリケーションのために用いられるようになるであろう。これが実現されると、ツールは、その使用法を簡単にすることが要求されるであろう。

インターネットとプライベートネットワークとの間の大きな違いの一つとしては、インターネットは非常に異質なネットワークであることが挙げられる。インターネットは、ネットワークのネットワークであり、別々の組織により操作されるネットワークの別々の部分では、大きな違いがある。これに対して、多くのプライベートネットワークは、比較的均質であるようにセットアップされ、ネットワークのアーキテクチャに関しては、プライベートネットワークのオペレータが大きな制御を及ぼしている。

マルチキャストネットワーク内の多数のエンドポイントは、別々のネットワークを用いて別々のレートでリンクされる可能性が高く、ネットワーク内の輻輳は、ネットワークの別々の部分ではそれぞれ違ってくるので、マルチキャストグループ内での加入リンクの容量に関する知識を得て、その容量でのパフォーマンスをテストすることができるのが望ましい。本発明による「マルチキャストネットワークプローブ」という特徴は、Mboneあるいはその他の大型異質マルチキャストネットワークをそのトラヒックソースから調査（プローブ）し、そのトラヒックソースから個々のリンクの容量を迅速に測定することができるように設計されている。

図6を参照すると、異質マルチキャストネットワーク（例えば、インターネットのMbone部分）は、５つのメンバーA～Eをもつマルチキャストグループを有している。ここで、グル

ープのそれぞれのメンバーは、それぞれ異なる容量のリンク（すなわち、異なるレートのリンク）により接続されている。グループのメンバーAは、64Kbps（キロビット／秒）のリンクによりネットワークに結びつけられており、Bは128Kbpsのリンクによって、Cは256Kbpsのリンクによって、Dは512Kbpsのリンクによって、そしてEは1024Kbpsのリンクによって結びつけられている。これらのリンク接続の性格は、サーバ（すなわち、トラヒックソース）には知られていない。なぜなら、インターネットへの接続は、異なる多数の速度のリンクを介しておこなわれうるからである。

デスティネーションへの情報のマルチキャスト転送をどのようにしておこなうかを最適のかたちで決定できるように、トラヒックソースが、デスティネーションへと至るそれぞれのリンクの特性を知ることが望ましい。もしアプリケーションがビデオ会議であるのなら、64KbpsのAへの品質は受け入れられないが、残りの者は、128Kbpsで参加できると決定されうる。同様に、もしアプリケーションがファイル転送であるのなら、DおよびEのグループは、512Kbpsの転送レートで動作するグループを構成できるが、A、BおよびCからなるグループは、ネットワークの容量を超えることなく、64Kbpsで動作可能である。

本発明によれば、ネットワークをプローブすることによって、遠隔のリンク容量を決定するメカニズムは、本願明細書で記載されたシステムおよびプロトコルである。メンバーA～Eから構成されるマルチキャストグループへのアナウンス／登録が、例えば、前セクションで記載した本発明の「マルチキャストピン」という特徴に基づいてコネクティビティを判定する（どのメンバーが実際にサーバに接続されているかを判定する）ための一手段として用いられた後、一連の小型テストファイルが、グループの個々のメンバーへとそれぞれ異なる速度で順次送られる。例えば、400個のフレームからなるテストファイルは、まず64Kbpsで送られた後、128Kbpsで送られ、次に256Kbpsで送られ、次に512Kbpsで送られ、そして最後に1024Kbpsで送られうる。クライアントの否定応答確認は、リンク上にその他のトラヒックがないものとすると、以下の表1に示すようにサーバで受信され、格納される。

送信速度	Aの NAK 数	Bの NAK 数	Cの NAK 数	Dの NAK 数	Eの NAK 数
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

表1－400フレームのテストファイルのテスト結果

表1を参照すると、速度64Kbpsでの第１のランは、グループのメンバーの誰に対しても否定応答確認（つまり、NAKsまたはNaks）はゼロという結果である。なぜなら、すべてのリンクは64Kbps以上をサポートしているからである。

第２のランは、第１のランの２倍である128Kbpsである。この第２のランでは、クライアントAは200個のNAKをもっている。つまり、全フレームの半分が失われたことを意味している。このことは、クライアントAの速度が、64Kbpsである（すなわち、（（400－200）／400）×128Kbps＝64Kbps）ことを意味している。クライアントB～Eは、第２のランでは失ったフレームがないことを示している。よって、これらのクライアントのそれぞれの速度は、少なくとも128Kbpsである。

第３のランでは、転送速度は256Kbpsであり、クライアントAおよびBは、それぞれ300個および50個のフレームを失ったことを示している。よって、この第３のランから、クライアントAの速度は、64Kbps（すなわち、（（400－300）／400）×256Kbps＝64Kbps）であることが分かる。これは、第２のランによる測定結果を裏書きしている。また、第３のランでは、クライアントBの速度は、128Kbps（すなわち、（（400－200）／400）×256Kbps=1

28Kbps）である。クライアントC～Eは、第３のランではエラーがなかったことを示している。よって、これらのクライアントはそれぞれ、少なくとも256Kbpsの速度で動作している。
第４のランでは、転送速度は512Kbpsである。クライアントAは、350個のフレームを失ったことを示している。よって、（（400－350）／400）×512Kbpsつまり64Kbpsと測定され、この結果はこれまでの測定と一致する。クライアントBは、300個のフレームを失ったことを示している。よって、（（400－300）／400）×512Kbpsつまり128Kbpsと測定され、この結果もこれまでのランと一致する。クライアントCは、200個のフレームを失ったことを示している。よって、（（400－200）／400）×512Kbpsつまり256Kbpsと測定される。
第５のランでは、転送速度は1024Kbpsである。クライアントAは、375個のフレームを失ったことを示している。よって、これまでと同様に（（400－375）／400）×1024Kbpsつまり64Kbpsと測定される。クライアントBは、（（400－350）／400）×1024Kbpsつまり128Kbpsと測定され、クライアントCは、（（400－300）／400）×1024Kbpsつまり256Kbpsと測定される。クライアントDは、（（400－200）／400）×1024Kbpsつまり512Kbpsと測定される。クライアントEには、脱落がない。このことは、その速度が少なくとも1024Kbpsであることを意味している。
よって、これら５つのランのそれぞれについて、ある与えられたリンクの容量は、以下の方程式により求められる。
（（送られたフレーム数－否定応答確認数）／送られたフレーム数）×送信速度＝リンク容量
このテスト技術もまた、リンク上のトラヒックを考慮に入れることになる。例えば、もしテストがおこなわれている間に物理リンクが256Kbpsであり、リンク上に128Kbpsのトラヒックがあるのなら、測定結果は、128Kbpsの容量、すなわちトラヒックが考慮される時の残りの容量となる。
これらのテストを実施するためのソフトウェアは、ソースがそれぞれのクライアントに対するリンク速度を知っているとすれば、リンクの品質をテストするのに用いることもできる。例えば、図6では、リンク速度は知ることができるので、フレームエラーレートを求めるためには、比較的長いテストパターンを用いてリンクをテストするのが望ましい。例えば、100,000フレームのテストファイルを、メンバーA～Eから構成されるグループに64Kbpsで送ればよい。伝送レートおよびNAKは、ソースに格納され、それぞれのクライアントからのNAK数は、それぞれのリンクの品質（すなわち、フレームエラーレート）を測定することを可能にする。Aは、最も重い負荷がかけられるリンクであるので品質が最悪になり、Eは、最小の負荷がかけられるリンクであるので最高になるだろうと予測されうる。しかし、その他のファクタがそれ以外の結果をもたらすこともある。同様に、より高速のリンクにより重い負荷を及ぼすためには、速度を増加させ、過剰な負荷のかけられたリンクをグループから削除してもよい。
よって、本発明による「マルチキャストネットワークプローブ」という特徴を用いれば、たとえ個々のリンクの容量が未知であっても、個々のリンクの容量を迅速に測定することができる。また、もしリンク速度がサーバには既知であるのなら、本発明のこの特徴は、それぞれのリンクの品質を判定する（すなわち、それぞれのリンクのフレームエラーレートを求める）のに用いることができる。
本発明のこの特徴によれば、マルチキャストグループのメンバーたちのコネクティビティが、まず、前述した「アナウンス／登録」フェーズを通ることにより判定される。すなわち、最初のステップは、グループ内のどのメンバーがサーバに接続されているかを判定することである。いったん接続されているメンバーがわかれば、テストファイル転送を開始して、テストファイルをそれぞれのメンバーに送り、その結果（すなわち、個々のグループメンバーについての否定応答確認の個数）を記録するサーバにより、リンク速度または品質を求めることができる。
速度グループ
サーバを複数のクライアントにインタフェースする多種多様なリンクそれぞれの容量、速

度、あるいは帯域幅（例えば、前セクションに述べた「マルチキャストネットワークプローブ」という特徴により利用可能となる）がわかれば、これらの速度のリストが、サーバにより格納されうる。その後、このリストは、リンク速度に基づいて複数のクライアントグループを生成あるいは規定するのに用いることができる。例えば、２つの速度グループがあり、その一方は、可能な最高速度が64Kbpsであるリンク（つまり、実効リンク）上でサーバに接続されているクライアントを含んでおり、他方は、可能な最高速度が1024Kbpsであるリンク（つまり、実効リンク）上でサーバに接続されているクライアントを含んでいることがある。よって、第２のグループは、第１のグループよりもはるかに高速である。特定の受け手がどの速度グループに属しているかということは、その受け手へのデータ転送に影響を及ぼす。本発明による最初のデータ転送パスのあいだ、第２の、より高速なグループ内のそれぞれの受け手には、サーバにより複数のフレームのすべてが送られるが、第１の、より低速なグループ内のそれぞれの受け手には、第２のグループに送られた16個のフレーム中の1個（1/16）が送られるにすぎない。このことは、第１のパスの後、サーバは、第２グループの受け手にはすべてのフレームを送っているが、第１のグループには、それらのフレームの全個数中の16分の1しか送っていないことを意味している。第１のグループにまだ送られていない残りのフレーム（つまり、全フレーム中の15/16）は、その後、後続するパスで第１グループの受け手に送られる。要点は、いったん、サーバにグループの各メンバーの容量がわかれば、サーバは、データの転送を調整することによって、より大容量のリンクを活用し、そのグループへのデータの転送を遅くすることがないという点にある。

否定応答確認収集

前述したように、本発明によりファイルを受信できるクライアントの数は、何千にも及ぶことがある。よって、サーバにより保守されるクライアントステータスリストのエントリ数も、何千にもなることがある。本発明によるファイル転送は、さらにその数量を高めることができる。例えば、何千もの受け手／クライアントの代わりに、何百万もの受け手／クライアントにファイルを送るように、スケーリングすることができる。好ましい実施の形態では、これらのクライアントすなわち受け手は、複数のクライアントから構成されるマルチキャストグループのメンバーである。

このスケーリングに関する特徴は、グループ内のクライアントの数があまりにも大きくなった時に発生する可能性のある問題を避ける一助になる。この問題は、多数のクライアントが否定応答確認をファイルの送り手（例えば、サーバ）に送り返し、送り手が程良い長さの時間の間に処理可能な限度を超えている個数の否定応答確認で、送り手を事実上、閉塞している時に起こる。これにより、送り手のパフォーマンスは低下する。なぜなら、送り手は、否定応答確認を受信し、処理するのに膨大な量の時間を費やす必要があり、その他の義務に従事することができないからである。このことが、送り手に戻るリンクを停滞させ、これらの否定応答確認のトラヒックにより渋滞することになる。

この問題に対する解決策は、「否定応答確認の収集」である。これにより、ファイルの送り手／サーバ20に渋滞を起こすことなく、クライアント／受け手の数を数千から数百万へと飛躍的に増やすことが逆に可能となる。この収集という特徴によれば、いくつかのクライアントあるいはその他のネットワークノードが「複製点」として作用し、他のクライアントからのブロック否定応答確認を収集する。好ましい実施の形態では、これらの複製点（RP）はルータである。図7を参照すると、米国全体にわたって５つのRPが示されている。また、それぞれのRPから発している線は、そのRPに接続されている１以上のクライアントを表している。例えば、RP100は、1200のクライアントを傘下に入れており、RP102は、900のクライアントを傘下に入れており、RP104は、100のクライアントを傘下に入れており、RP106は、800のクライアントを傘下に入れており、RP108は、500のクライアントを傘下に入れている。サーバすなわちソース20は、米国内の別の場所に位置している。RP100は、それに加入しているか、または接続されている（例えば、1200の）クライアントからのブロック否定応答確認のすべてを収集する。その他のRP102、104、106および108も、それらに加入しているクライアントに対して同じことをする。それぞれのRPについて、それ

に加入しているクライアントのすべてからすべてのブロック否定応答確認を収集した後、そのRPがサーバ20、またはサーバ20へと向かうチェーンの別のRPへとただ１つの応答確認メッセージを送る。その１個のメッセージが、そのRPに加入しているすべてのクライアントからのブロック否定応答確認のすべてを含んでいる。サーバ20が、最終的にこれらの収集されたブロック否定応答確認メッセージを複数のRPから受信すると、サーバは、次のパスで否定応答確認されたフレームのすべてを送り返す。これらのRPは、これらの後続パスフレームを受信し、それらのフレームを、チェーン内の適切なクライアントまたはその他のRPへと送り出す責任をもつ。それらのクライアントまたはRPは、その後、それらのフレームをチェーン内の適切なクライアントまたはその他のRPへと送り出す。以下、この操作が繰り返される。

ここに記載されたことに関する変更、改変およびその他の実現形態は、クレームされている本発明の精神および範囲から離れることなく、当業者には着想可能であろう。したがって、本発明は、以上に記した例示的な説明によってではなく、以下の請求の範囲によって規定されるべきである。

【図１】

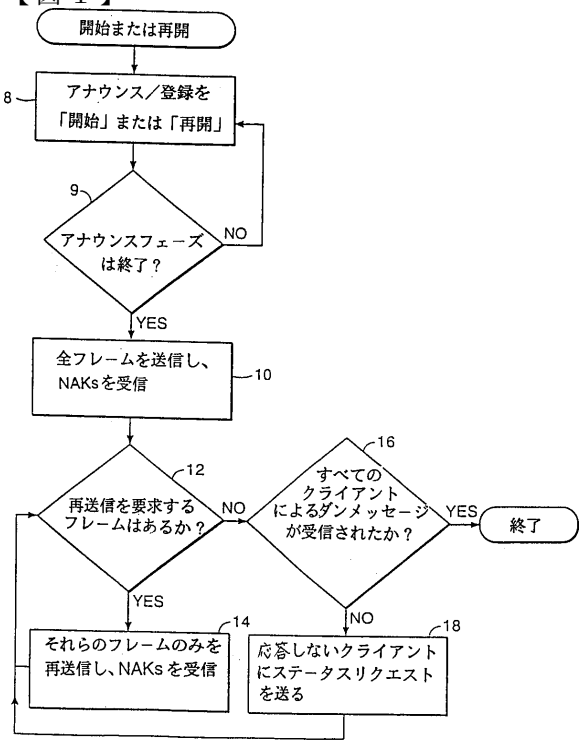


FIG. 1

【図２】

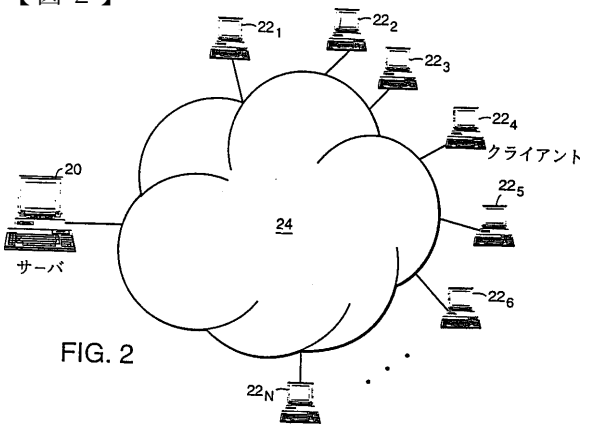


FIG. 2

【図３】

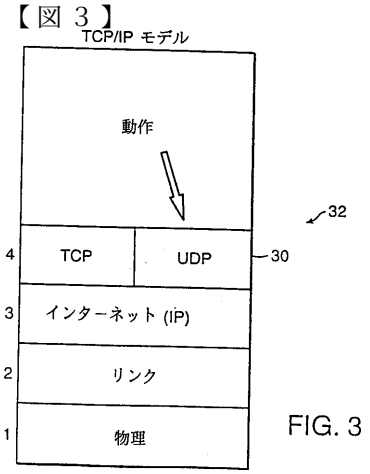


FIG. 3

【図４】

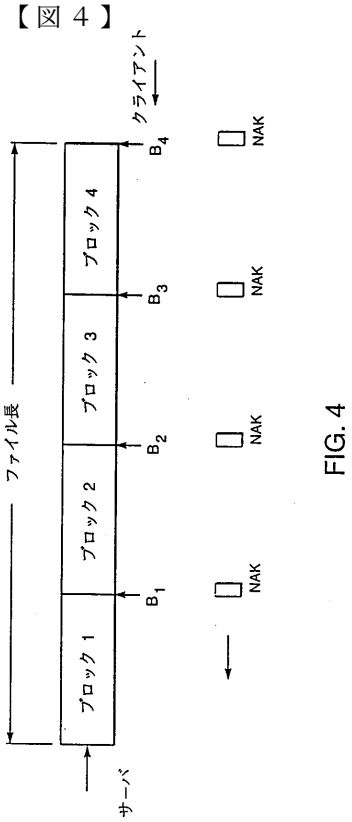


FIG. 4

【図５】

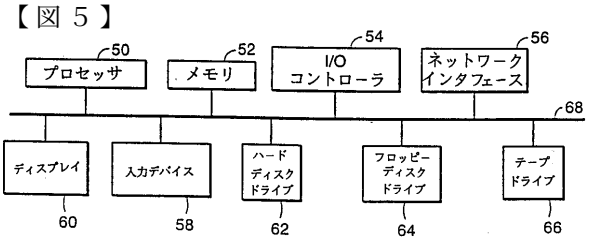
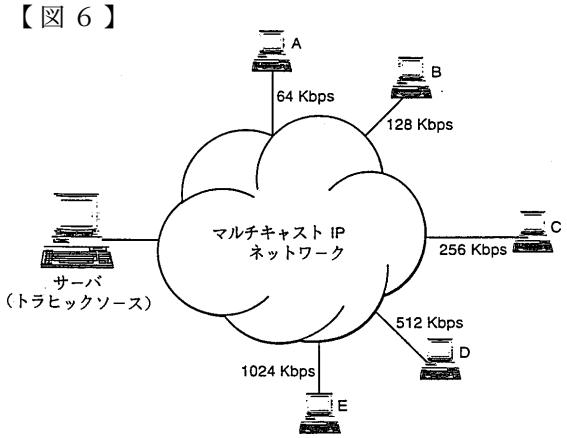


FIG. 5

【図６】



異質マルチキャストIPネットワーク
（例：Mbone）

FIG. 6

【図７】

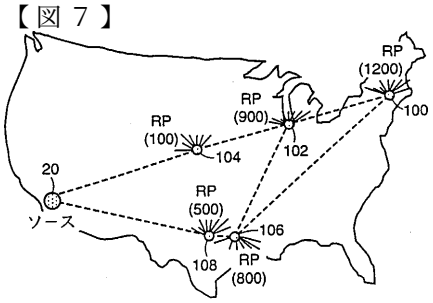


FIG. 7

【図８】

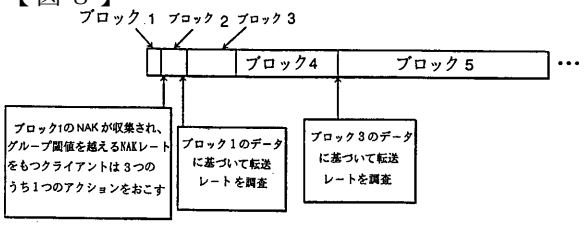


FIG. 8

【図９】

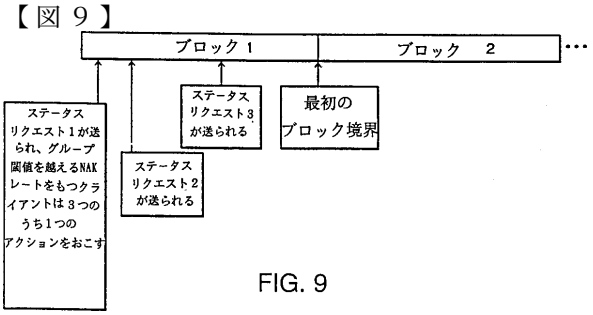


FIG. 9

フロントページの続き

(72)発明者 ロバートソン，ケリー
アメリカ合衆国 マサチューセッツ ０１９５０，ニューバリーポート，ショア ロード １
(72)発明者 ケイツ，ケニース
アメリカ合衆国 ニューハンプシャー ０３０７９，セイラム，アッカーマン ストリート ４７
(72)発明者 ホワイト，マーク
アメリカ合衆国 マサチューセッツ ０１７７８，ウェイランド，コンコード ロード ３１５

審査官 石井 研一

(56)参考文献 特開平０２－２７２９７５（ＪＰ，Ａ）
特開平０４－２０７４３０（ＪＰ，Ａ）
特開平０６－２５２８９６（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.^{7}，ＤＢ名)
H04L 12/56
H04L 1/18
H04L 12/18
H04L 12/28