(19)日本国特許庁(JP)　(12)公表特許公報(A)　(11)特許出願公表番号
特表2006-505215
(P2006-505215A)
(43)公表日　平成18年2月9日(2006.2.9)

(51)Int.Cl.		F I			テーマコード（参考）
H04L 12/56	(2006.01)	H04L	12/56	300A	5K030
G06F 13/00	(2006.01)	G06F	13/00	520B	

審査請求 未請求　予備審査請求 未請求　（全 38 頁）

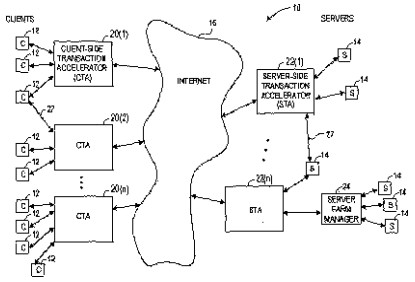
(21)出願番号　特願2004-550165 (P2004-550165)
(86)(22)出願日　平成15年10月28日 (2003.10.28)
(85)翻訳文提出日　平成17年7月4日 (2005.7.4)
(86)国際出願番号　PCT/US2003/034232
(87)国際公開番号　WO2004/042508
(87)国際公開日　平成16年5月21日 (2004.5.21)
(31)優先権主張番号　10/285,315
(32)優先日　平成14年10月30日 (2002.10.30)
(33)優先権主張国　米国 (US)

(71)出願人　505163486
リバーベッド　テクノロジー　インコーポレーティッド
アメリカ合衆国　カリフォルニア州　サンフランシスコ　スイート　４１０　セカンド　ストリート　５０１
(74)代理人　100102978
弁理士　清水　初志
(74)代理人　100128048
弁理士　新見　浩一
(72)発明者　マッカン　スティーブン
アメリカ合衆国　カリフォルニア州　バークリー　エル　カミノ　リアル　５４

最終頁に続く

(54)【発明の名称】クライアント－サーバ通信システムのトランザクション・アクセルレータ

(57)【要約】
トランザクション加速が可能なネットワークでは、トランザクションを加速する場合、クライアントは、要求をクライアント側トランザクション・ハンドラに送り、クライアント側トランザクション・ハンドラは要求をサーバ側トランザクション・ハンドラに送り、サーバ側トランザクション・ハンドラは要求またはその表現を、要求に応答するサーバに与える。サーバは、応答をサーバ側トランザクション・ハンドラに送信し、サーバ側トランザクション・ハンドラは応答をクライアント側トランザクション・ハンドラに転送し、クライアント側トランザクション・ハンドラは応答をクライアントに与える。トランザクションは、サーバ側トランザクション・ハンドラがアクセスできる持続セグメント記憶装置およびクライアント側トランザクション・ハンドラがアクセスできる持続セグメント記憶装置に、トランザクションで使用されるデータのセグメントを記憶することにより、トランザクション・ハンドラによって加速される。データをトランザクション・ハンドラ間で送信する場合、送信側トランザクション・ハンドラは、送信すべきデータのセ



【特許請求の範囲】
【請求項１】
クライアントがサーバとのトランザクションを開始し、ネットワークが、要求メッセージをクライアントからサーバに、応答メッセージをサーバからクライアントに転送し、要求メッセージおよび応答メッセージのうちの少なくとも1つが、ネットワーク上で送信側から受信側に転送すべきペイロードを含むネットワークにおいて、トランザクションを加速する方法であって、
ペイロードを、少なくとも1つのセグメント、セグメント化されないデータの1つもしくは複数の記号、または少なくとも1つのセグメントとセグメント化されないデータの1つもしくは複数の記号との組合せにセグメント化する段階と、
各セグメントごとに、セグメント・データをペイロードで参照無しセグメントとして送信するか参照付きセグメントとして送信するかを判定する段階と、
各参照付きセグメントごとに、少なくともいくつかのセグメント・データを、置き換えられるデータの参照で置き換え、セグメントの参照に関連する置き換えられた参照データを記憶する段階と、
ペイロードを、少なくとも1つのセグメント参照および参照無しデータの記号を含む加速されたペイロードとして送信側から受信側に送信する段階と、
加速されたペイロードをネットワーク上で受信する段階と、
加速されたペイロード内に、もしあればセグメント参照を識別する段階と、
加速されたペイロード内のセグメント参照を、受信側から利用できるセグメント・データで置き換え、再構成されたペイロードを形成する段階と、
再構成されたペイロードを、転送されたペイロードとして受信側に提供する段階とを含む方法。
【請求項２】
2つ以上のトランザクション・アクセルレータをネットワーク内に配置する段階と、
第1のトランザクション・アクセルレータにある送信側サーバまたは送信側クライアントからペイロードを受信する段階と、
加速されたペイロードを第1のトランザクション・アクセルレータから第2のトランザクション・アクセルレータへ送信する段階と、
加速が送信側の送信プロトコルおよび受信側の受信プロトコルに対して透過的になるように、再構成されたペイロードを第2のトランザクション・アクセルレータから受信側クライアントまたは受信側サーバに送信する段階とをさらに含む、請求項1記載の方法。
【請求項３】
受信側に関連するトランザクション・アクセルレータにおいて、参照付きセグメントのセグメント・データがペイロードの再構成に利用できないのはいつかを判定する段階と、
送信側に関連するトランザクション・アクセルレータにセグメント・データを要求する段階とをさらに含む、請求項1記載の方法。
【請求項４】
セグメント・データがセグメント参照の一部として記憶されているとき、セグメント・データをその反転可能な関数を介して変換し、この変換の結果を記憶する段階と、
セグメント・データをセグメントを参照解除する際に用いるとき、反転可能な関数の逆数を用いて、記憶されている結果を逆変換する段階とをさらに含む、請求項1記載の方法。
【請求項５】
反転可能な関数は、順方向誤り補正関数、暗号関数、および署名関数のうちの1つまたは複数である、請求項4記載の方法。
【請求項６】
クライアントがサーバとのトランザクションを開始し、ネットワークが、要求メッセージをクライアントからサーバに、応答メッセージをサーバからクライアントに転送し、要求メッセージおよび応答メッセージのうちの少なくとも1つが、ネットワーク上で送信側

から受信側に転送すべきペイロードを含むネットワークにおける、トランザクション・アクセルレータであって、
送信すべきメッセージのペイロードを変換し、
a）ペイロードを、少なくとも1つのセグメント、セグメント化されないデータの1つもしくは複数の記号、または少なくとも1つのセグメントとセグメント化されないデータの1つもしくは複数の記号との組合せにセグメント化するセグメンタ、および
b）各セグメントのセグメント・データを参照無しセグメントとして表示するかそれとも参照付きセグメントとして表示するかを判定するリプレーサを含むトランザクション変換器と；
参照付きセグメントのセグメント・データおよびセグメント参照を記憶する持続セグメント・ストアと；
リモート・トランザクション・アクセルレータのトランザクション変換器からの加速されたペイロードを逆変換し、
a）セグメント参照がアクセルレータ・ペイロード内のどこに現れるかを判定するトークナイザ、および
b）持続セグメント・ストアのセグメント・データを、トークナイザによって検出された各セグメント参照で置き換えるデレファレンサ
を含む逆トランザクション変換器と；
持続セグメント・ストアに存在しないセグメントのセグメント参照に出会ったときにデレファレンサの必要に応じて参照を変換する参照リゾルバとを含む、トランザクション・アクセルレータ。
【請求項７】
クライアントがサーバとのトランザクションを開始し、ネットワークが、要求メッセージをクライアントからサーバに、応答メッセージをサーバからクライアントに転送し、要求メッセージおよび応答メッセージのうちの少なくとも1つが、ネットワーク上で送信側から受信側に転送すべきペイロードを含むネットワークにおける改良であって、
a）サーバに供給される要求メッセージを受信し、サーバからの再構成された応答メッセージをクライアントに関係付けるようにクライアントに結合されたプロキシ；
b）プロキシから受信された要求メッセージを変換し、
i）要求メッセージ・ペイロードを、少なくとも1つのセグメント、セグメント化されないデータの1つもしくは複数の記号、または少なくとも1つのセグメントとセグメント化されないデータの1つもしくは複数の記号との組合せにセグメント化するセグメンタと、
ii）各セグメントのセグメント・データを参照無しセグメントとして表示するか参照付きセグメントとして表示するかを判定するリプレーサとを含むトランザクション変換器、
c）要求メッセージ・ペイロードの参照付きセグメントのセグメント・データおよびセグメント参照を記憶し、それぞれの異なるセグメント・データを有するセグメントがそれぞれの異なるセグメント参照を有し、それぞれの異なるセグメント参照が、セグメント参照が作成されたトランザクションと無関係であってよい持続セグメント・ストア、ならびに
d）応答メッセージが加速されたときに応答メッセージ・ペイロードを再構成された応答メッセージに逆変換し、
i）セグメント参照が応答メッセージ・ペイロード内のどこに現れるかを判定するトークナイザ、および
ii）持続セグメント・ストアのセグメント・データを、トークナイザによって検出された各セグメント参照で置き換えるデレファレンサ
を含む逆トランザクション変換器
を含むクライアント側トランザクション・アクセルレータと；
a）クライアントに供給される応答メッセージを受信し、クライアントからの再構成

された要求メッセージをサーバに関係付けるようにサーバに結合されたプロキシ、
b）プロキシから受信された応答メッセージを変換し、
i）応答メッセージ・ペイロードを、少なくとも1つのセグメント、セグメント化されないデータの1つもしくは複数の記号、または少なくとも1つのセグメントとセグメント化されないデータの1つもしくは複数の記号との組合せにセグメント化するセグメンタ、および
ii）各セグメントのセグメント・データを参照無しセグメントとして表示するか参照付きセグメントとして表示するかを判定するリプレーサを含むトランザクション変換器、
c）応答メッセージ・ペイロードの参照付きセグメントのセグメント・データおよびセグメント参照を記憶し、それぞれの異なるセグメント・データを有するセグメントがそれぞれのセグメント参照を有し、それぞれの異なるセグメント参照が、セグメント参照が作成されたトランザクションと無関係であってよい持続セグメント・ストア、ならびに
d）要求メッセージが加速されたときに要求メッセージ・ペイロードを再構成された要求メッセージに逆変換し、
i）セグメント参照が要求メッセージ・ペイロード内のどこに現れるかを判定するトークナイザ、および
ii）持続セグメント・ストアのセグメント・データを、トークナイザによって検出された各セグメント参照で置き換えるデレファレンサ
を含む逆トランザクション変換器
を含むサーバ側トランザクション・アクセルレータであって、
要求メッセージ・ペイロードまたは応答メッセージ・ペイロードのいずれかにおける少なくとも一つのセグメントが参照付きセグメントとして送信される、サーバ側トランザクション・アクセルレータ
とを含む改良。

【請求項８】
クライアント側持続セグメント・ストアに存在しないセグメントのセグメント参照に出会ったときにクライアント側デリファレンサの必要に応じて参照を変換するクライアント側参照リゾルバと、
サーバ側持続セグメント・ストアに存在しないセグメントのセグメント参照に出会ったときにサーバ側デリファレンサの必要に応じて参照を変換するサーバ側参照リゾルバとをさらに含む、請求項7記載の改良。

【請求項９】
クライアント側トランザクション・アクセルレータとサーバ側トランザクション・アクセルレータの一方は第1のトランザクション・アクセルレータであり、クライアント側トランザクション・アクセルレータとサーバ側トランザクション・アクセルレータの他方は第2のトランザクション・アクセルレータであり、改良は、
第2のトランザクション・アクセルレータのセグメント要件を予想する、第1のトランザクション・アクセルレータにある手段と、
第2のトランザクション・アクセルレータのデリファレンサが予想されるセグメントを必要とする前に、このような予想されるセグメントを、第1のトランザクション・アクセルレータから第2のトランザクションの持続セグメント・ストアに転送する手段とをさらに含む、請求項7記載の改良。

【請求項１０】
ネットワーク上で協働する1組のトランザクション・アクセルレータの各トランザクション・アクセルレータに位置し、実際上境界のない識別子スペースから選択されるセグメント参照識別子が1組のトランザクション・アクセルレータ全体にわたって一意になるようにセグメント参照識別子を割り当てる手段と、
セグメント識別子を圧縮して、1回または複数回のデータ記憶およびデータ送信に使用される圧縮されたセグメント識別子を形成する手段とをさらに含む、請求項7記載の改良

。
【発明の詳細な説明】
【技術分野】
【０００１】
関連出願の相互参照
「Content-Based Segmentation Scheme for Data Compression in Storage and Transmission Including Hierarchical Segment Representation」 [Attorney Docket No: 021647-000200US]（以下「McCanne II」）という名称の米国特許出願第10/285,330号は、本出願と同じ日付で出願されており、参照としてすべての目的で本明細書に組み入れられる。
【０００２】
発明の背景
本発明は概して、データを限られた帯域幅チャネルを効率的に移動させるシステムに関し、特に、あるデータを求める要求に応答し、データがその要求に応答して未処理で送信される場合よりも高速に、限られたチャネル上でデータを利用可能にすることに関する。
【背景技術】
【０００３】
高速接続上でうまく動作する多くのアプリケーションおよびシステムは、より低速の接続上で動作するように適合させる必要がある。たとえば、ローカル・エリア・ネットワーク（LAN）上でファイル・システムを動作させるとうまく働くが、ファイルにアクセスする必要のあるクライアントからそのファイルを提供するファイル・サーバまでのパス全体にわたってLANなどの高速リンクが利用できないファイルにアクセスしなければならなくなることが少なくない。eメール・サービス、計算サービス、マルチメディア、テレビ会議、データベース問合せ、オフィス・コラボレーションのような他のネットワーク・サービスについても設計上の同様の問題が存在する。
【０００４】
ネットワーク化されたファイル・システムでは、たとえば、ある場所でアプリケーションによって使用されているファイルを別の場所に記憶することができる。通常、ある組織および／または地理的領域全体にわたってネットワーク化されたコンピュータで動作する数人かのユーザは、ファイル・システムに記憶されているファイルまたは数組のファイルを共用する。ファイル・システムは、ユーザのうちの1人に近い場合もあるが、通常、大部分のユーザから遠い。しかし、ユーザはファイルが自分のサイトに近くにあるように見えることを望む。
【０００５】
本明細書では、「クライアント」は一般に、データまたは動作を要求するコンピュータ、コンピューティング・デバイス、周辺装置、電子機器などを指し、一方、「サーバ」は一般に、1つまたは複数のクライアントからのデータまたは動作を求める要求に応答して動作するコンピュータ、コンピューティング・デバイス、周辺装置、電子機器などを指す。
【０００６】
要求は、コンピュータ、コンピューティング・デバイス、周辺装置、電子機器などの動作を求める要求であっても、クライアントによって実行または制御されているアプリケーションを求める要求であってもよい。一例として、コンピュータの外部に記憶されている文書を必要とし、ネットワーク・ファイル・システム・クライアントを用いてネットワーク上でファイル・サーバに要求を出す文書作成プログラムを動作させるコンピュータが挙げられる。他の例として、プリント・サーバ、処理サーバ、制御サーバ、機器インタフェース・サーバ、I/O（入出力）サーバのような、それ自体が動作を実行するサーバに対する、動作を求める要求がある。
【０００７】
要求は、要求されたデータを供給するかもしくは要求された動作を実行する応答メッセージ、または要求が失敗したかもしくは不適切であったことを示す監視システムへのエラ

ー・メッセージや警告のような、要求を満たすことができないことを示す応答メッセージによって満たされることが多い。サーバは、要求を遮断したり、転送したり、変換したりすることもでき、さらに要求に応答することもしないこともある。

【０００８】

場合によっては、通常サーバとみなされるオブジェクトが、クライアントとして働き要求を出すことができ、通常クライアントとみなされるオブジェクトが、サーバとして働き要求に応答することができる。さらに、単一のオブジェクトが、他のサーバ／クライアントに対するサーバとクライアントの両方、またはそれ自体に対するサーバとクライアントの両方であってよい。たとえば、デスクトップ・コンピュータは、データベース・クライアントおよびデータベース・クライアント用のユーザ・インタフェースを実行することができる。デスクトップ・コンピュータ・ユーザがデータベース・クライアントを操作してデータを求めるようを出させる場合、データベース・クライアントは要求をおそらくデータベース・サーバに発行する。データベース・サーバが同じデスクトップ・コンピュータ上で実行される場合、デスクトップ・コンピュータは実際上、それ自体に要求を出す。本明細書では、クライアントとサーバが異なり、ネットワーク、物理的な距離、セキュリティ手段、およびその他の障壁によって分離されることが多いが、これはクライアントおよびサーバの必要な特徴ではないことを理解されたい。

【０００９】

場合によっては、クライアントとサーバは必ずしも排他的なものではない。たとえば、ピア・ツー・ピア・ネットワークでは、あるピアが別のピアに要求を出すことができるが、そのピアに応答することもできる。したがって、語「クライアント」および「サーバ」は通常、本明細書ではそれぞれ「要求」を出す動作主および「応答」を与える動作主として使用されるが、これらの要素が、クライアント－サーバの例では明確に表されない他の役割を果たすことができることを理解されたい。

【００１０】

一般に、要求－応答サイクルを「トランザクション」と呼ぶことができ、所与のトランザクションでは、あるオブジェクト（物理的、論理的、および／または仮想）をそのトランザクションの「クライアント」と呼ぶことができ、他のあるオブジェクト（物理的、論理的、および／または仮想）をそのトランザクションの「サーバ」と呼ぶことができる。

【００１１】

クライアント－サーバ・トランザクションは、パケット・ネットワークを横切ってクライアントとサーバとの間を直接流れることが多いが、環境によっては、これらのトランザクションを遮断し、「プロキシ」と呼ばれる転送レベルまたはアプリケーション・レベルの装置を通じて転送することができる。この場合、プロキシは、クライアントの接続の末端であり、クライアントのためにサーバとの別の接続を開始する。または、プロキシは、サーバに接続された1つまたは複数の他のサーバに接続される。各プロキシは、トランザクションがクライアントからサーバに流れ、サーバからクライアントに流れるときにトランザクションを転送、修正、または他の方法で変換することができる。プロキシの例には、（1）サーバへのアクセスを制御することによりキャッシングによって性能を向上させるかまたはセキュリティを向上させるWebプロキシ、（2）メールをクライアントから別のメール・サーバに転送するメール・リレー、（3）変換されたDNS名などをキャッシングするDNSリレーが含まれる。

【００１２】

本明細書では、「近い」、「遠い」、「ローカル」、および「リモート」という用語は物理的な距離を指すことができるが、通常、有効距離を指す。2つのコンピュータ、コンピューティング装置、サーバ、クライアント、周辺装置などの間の有効距離は、少なくとも概ね、2つのコンピュータの間でデータを得ることの困難さの尺度である。たとえば、ファイル・データがそのファイル・データを使用するコンピュータ・プロセッサに直接接続されたハード・ドライブ上に記憶されており、接続が専用高速バスによるものである場合、ハード・ドライブとコンピュータ・プロセッサは実際上互いに「近い」が、ハード・

ドライブとコンピュータ・プロセッサとの間のトラフィックが低速バス上のトラフィックであり、データを遮断する可能性のあり介在するイベントの数がより多い場合、ハード・ドライブとコンピュータ・プロセッサは遠く離れていると言われる。

【００１３】

必ずしも物理的距離は有効距離に比例するわけではない。たとえば、何マイルもの高品質・高帯域幅光ファイバによって分離されたファイル・サーバとデスクトップ・コンピュータは、数フィート分離され雑音の多い環境で無線接続を介して結合されたファイル・サーバとデスクトップ・コンピュータと比べてより短い有効距離を有する。

【００１４】

一般に、有効距離が長い場合、有効距離がそれよりも短いという印象を与えるにはより多くの工夫が必要である。この印象を与えるために多くの開発がなされている。たとえば、帯域幅が限られているために有効距離が長くなるときは、圧縮またはキャッシングによってその制限を軽減することができる。圧縮は、いくつかのデータ・ビットをそれよりも少ないビットを用いて表し、かつそれを、たいていの場合、元のビットまたは元のビットの少なくとも十分な近似を圧縮プロセスを反転させたプロセスから回復できるように行うプロセスである。キャッシングは、すでに送信されている結果を、ユーザが、その結果を再び要求し、結果を元のプロバイダから得る必要がある場合よりも高速にキャッシュから受信することを期待して記憶するプロセスである。

【００１５】

圧縮は、限られた帯域幅をより効率的に使用するのを可能にし、呼出し時間を短くすることができるが、場合によっては、呼出し時間が改善されないこともある。クライアント－サーバ・トランザクションに関する呼出し時間は、データを求める要求が出されてから要求されたデータが受信されるまでの遅延の尺度である。要求が出された後でデータを圧縮する時間が必要であり、かつデータが受信された後で解凍する時間が必要である場合、圧縮が呼出し時間を延ばすこともある。これは、データが要求が出される前に圧縮できる場合には改善することができるが、データが圧縮よりも前に得られるとは限らない場合や、要求を満たすためのデータの量が使用される可能性の高いデータの量と比べて多すぎる場合には実現不能である場合がある。

【００１６】

キャッシングも、有効距離を短くするのをある程度助けるが、状況によってはそれほど有効ではない。たとえば、単一のプロセッサが、それが制御するメモリからデータを受信しており、かつメモリからプロセッサ命令を読み取る場合のように、それを反復的に行っている場合、キャッシングはプロセッサのタスクを大幅に加速することができる。代表的なキャッシュ構成では、要求側はあるメモリ、装置などにデータを要求し、結果は、要求側に与えられ、データを供給する元の装置よりも短い応答時間を有するキャッシュに記憶される。次いで、要求側がそのデータを再び要求すると、データが依然としてキャッシュ内にある場合、キャッシュは、元の装置が返すことができたであろう時間よりも前に要求に対する応答でそのデータを返すことができ、要求はずっと短時間で満たされる。

【００１７】

キャッシングには難点があり、そのうちの1つは、データが送信元で変更されることがあり、その場合、キャッシュが「古い（stale）」データを要求側に供給することである。キャッシングに関する他の問題は、データの送信元がデータの使用法を追跡することを望む場合があり、かつ送信元ではなくキャッシュから供給されたデータの使用を認識しないことである。たとえば、Webサーバが、そのWebサーバから「アクセス可能な（pointed to）」Webブラウザを実行するいくつかのコンピュータに対してリモート位置にある場合、Webブラウザは、Webページを見るときにそのページをそのサイトからキャッシングし、そのWebページを再びダウンロードする際に起こる可能性のある遅延を回避する。これは多くの場合性能を向上させ、Webサーバ上の負荷を削減するが、Webサーバ・オペレータは、「ページ・ビュー」の総数を知ることを試みることがあるにもかかわらず、キャッシュによって実行される「ページ・ビュー」は無視する。場合によっては、インターネット・

サービス・プロバイダは、ブラウザに対してリモート位置のキャッシュを使用し、キャッシングされているコンテントを多数のブラウザに与えることがあり、したがって、場合によっては、Webサーバ・オペレータには固有のユーザがまったく見えなくなる可能性がある。

【００１８】

さらに、Webキャッシングの基礎となる機構は、元のデータとキャッシングされたデータとの整合性について厳密でないモデルしか有していない。一般に、Webデータは、元のデータの変更とは無関係にトランザクションにおけるヒューリスティックスまたは経験に基づく期間にわたってキャッシングされる。このことは、キャッシングされたWebデータが、元のサーバと矛盾することがあり、このような矛盾が、Webサイト・オペレータ、サービス・プロバイダ、およびユーザによって、単に性能面の合理的な兼ね合わせとして許容されることを意味する。残念なことに、整合性に関して厳密でないこのモデルは、ネットワーク化されたファイル・システムのような一般的なクライアント－サーバ通信には完全に不適切である。クライアントがファイル・サーバと対話する際、整合性モデルは、このファイル・システムを使用するアプリケーションを適切に動作させるように全体的に正しくかつ正確でなければならない。

【００１９】

ネットワーク応答に対するいくつかの解決策は、ファイル・システムまたはネットワーク・レイヤにおける問題に対処する。提案されている1つの解決策は、Muthitacharoen, A., et al., 「A Low-Bandwidth Network File System」 in Proceedings of the 18th ACM Symposium on Operating Systems Principles (SOSP '01), pp. 174-187 (Chateau Lake Louise, Banff, Canada、2001年10月)(ACM SIGOPS Operating Systems Review, ACM Pressの第35巻第5号)に記載されているように、低帯域幅ネットワーク・ファイル・システムを使用することである。LBFSと呼ばれるこのシステムでは、クライアントは「全ファイル」キャッシングを使用し、それによって、ファイル・オープン動作時に、クライアントはファイル内のすべてのデータをサーバから取り込み、次いでファイル・データのローカルにキャッシュされたコピーを処理する。クライアントがファイルを変更した場合、これらの変更は、クライアントがファイルを閉じたときにサーバに伝搬される。これらの転送を最適化するために、LBFSはファイルの一部をハッシュで置き換え、受信側はこのハッシュをローカル・ファイル・ストアと共に使用してハッシュをファイルの元の部分に変換する。このようなシステムは、ファイル・システムに結合されており、一般に応答を改善する予定のクライアントおよびサーバを修正する必要がある。さらに、ハッシング方式は、比較的大きな（平均的な）サイズのブロックに対して作用し、ファイルが長時間にわたる微細な変更を受けるときには不十分である。最後に、LBFSは、設計上ネットワーク・ファイル・システム・プロトコルに密に結合される。LBFSは、他の種類のクライアント－サーバ・トランザクション、たとえば、eメール、Web、ストリーミング媒体などを最適化することも加速することもできない。

【００２０】

提案されている他の解決策はSpring, N., et al., 「A Protocol-Independent Technique for Eliminating Redundant Network Traffic」 in Proceedings of ACM SIGCOMM（2000年8月）によって示唆されている。この文献に記載されているように、繰り返されているストリングを識別し、繰り返されているストリングをネットワーク・リンクのいずれかの端部にある共用パケット・キャッシュから得られるトークンで置き換えることによって、最近送信されたパケットに類似のネットワーク・パケットのサイズを小さくすることができる。この手法は、有利であるが、いくつかの欠点を有する。この手法は個々のパケットに作用するに過ぎないので、得られる性能面の利益は、パケット・ペイロード・サイズとパケット・ヘッダとの比によって制限される（パケット・ヘッダは一般に、前述の技術を用いても圧縮できないため）。さらに、この機構は、パケット・レベルで実現されるため、ネットワークの、通信パスの2つの端部が装置を備えている領域に適用されるに過ぎない。この構成は実現するのが困難なことがあり、ある環境では実際的ではない。さらに

、比較的小さなメモリ・ベースのキャッシュを先入れ先出し交換方式と共に使用して（たとえば、大きなディスク・ベースのバッキング・ストアの助けなしで）ネットワーク・パケットをキャッシングすることによって、この手法の効果は、時間的にかなり制限された通信冗長性を検出し活用することに制限される。最後に、この手法は、（冗長）ネットワーク・トラフィックを生成するアプリケーションやサーバと結合されないので、データがどこで使用されるかを予測し、そのデータを、ネットワーク・トラフィックをさらに加速し最適化するファー・エンド・キャッシュに事前にステージングしておくことはできない。

【００２１】

ワイド・エリア・ネットワーク上で業務を展開する場合、上述の問題に対して理想的なものではないいくつかのパッチが実行されている。たとえば、企業によっては、応答性を維持するために購入する帯域幅を高めている。組織内の個人は、アドホックeメール・コラボレーション（1つのファイルを1人のユーザからより容易にアクセスできるようにするが、バージョン制御問題を引き起こし、全体的なネットワーク負荷を増大させる）を使用することによってローカルな解決策を試みるであろう。この問題を解決するための他の試みには、処理すべきデータのコピーをユーザが作成することや、読取り専用レプリカをリモート・サーバに転送（push）する。

【００２２】

上記の問題および既存の解決策に対する制限を考慮して、ネットワーク上のトランザクションのためにデータを転送する方法を向上させることができる。

【発明の開示】

【００２３】

発明の簡単な概要

トランザクション加速が可能なネットワークの態様では、トランザクションを加速する場合、クライアントは、要求をクライアント側トランザクション・ハンドラに送り、クライアント側トランザクション・ハンドラは要求をサーバ側トランザクション・ハンドラに送り、サーバ側トランザクション・ハンドラは要求またはその表現を、要求に応答するサーバに与える。サーバは、応答をサーバ側トランザクション・ハンドラに送信し、サーバ側トランザクション・ハンドラは応答をクライアント側トランザクション・ハンドラに転送し、クライアント側トランザクション・ハンドラは応答をクライアントに与える。トランザクションは、サーバ側トランザクション・ハンドラがアクセスできる持続セグメント記憶装置およびクライアント側トランザクション・ハンドラがアクセスできる持続セグメント記憶装置に、トランザクションで使用されるデータのセグメントを記憶することにより、トランザクション・ハンドラによって加速される。データをトランザクション・ハンドラ間で送信する場合、送信側トランザクション・ハンドラは、送信すべきデータのセグメントを、その持続セグメント記憶装置に記憶されているセグメントと比較し、置き換えるべきデータのセグメントと一致するかまたはほぼ一致するその持続セグメント記憶装置内のエントリの参照でデータのセグメントを置き換える。送信すべきデータは、クライアントからサーバに送信したり、サーバからクライアントに送信したり、ピアからピアに送信したりすることができる。この場合、受信側トランザクション・ストアは、セグメント参照をその持続セグメント記憶装置からの対応するセグメント・データで置き換えることによって、送信されたデータを再構成する。セグメントが参照されているが、受信側の持続セグメント・ストアに存在しない場合、受信側は、サイド・チャネルまたはセグメントの参照を送信するのに用いられるリンクを介して、存在しないセグメントを送信側に求める要求を発行することができる。各端部における持続セグメント記憶装置に、繰り返される可能性の高いセグメントが存在する場合、セグメントのこのような置き換えが行われることが多く、ネットワーク上の帯域幅の使用量がずっと少なくなり、したがって、トランザクションが加速される

【００２４】

トランザクション・アクセルレータは、クライアント側トランザクション・アクセルレ

ータが1つのクライアントとのみ対話し、サーバ側トランザクション・アクセルレータが1つのサーバとのみ対話するように専用であってよいが、トランザクション・アクチュエータは、複数のクライアントおよび／または複数のサーバを取り扱うことができる。複数のトランザクションを取り扱う場合、同じクライアントおよびサーバの場合でも、場合によっては異なるクライアントおよび場合によっては異なるサーバの場合でも、持続セグメント・ストアに記憶されているセグメントをそれぞれの異なるトランザクション、それぞれの異なるクライアント、および／またはそれぞれの異なるサーバに関係付けることができる。たとえば、トランザクション・アクセルレータが、所与のトランザクションを取り扱う際に、データのセグメントを見つけその持続セグメント・ストアに記憶した場合、そのデータ・セグメントの参照を異なるトランザクションで再び使用し、異なるクライアントまたは同じクライアントおよび異なるサーバまたは同じサーバに関係付けるか、またはまったく異なるクライアント・サーバ・アプリケーションに関係付けることができる。

【００２５】

いくつかの態様においては、トランザクション・アクセルレータの持続セグメント・ストアに他のトランザクション・アクセルレータから得たセグメント・データが存在し、したがって、トランザクションが行われる際、送信側において、参照との置換えに利用できるセグメントが増え、受信側において、参照からの再構成に利用できるセグメントが増える。

【００２６】

本発明の他の特徴および利点は、以下の詳細な説明および好ましい態様を鑑みて明らかである。

【００２７】

発明の詳細な説明

本発明は、本開示を読んだ後で明らかになる多数の用途を有する。本発明によるトランザクション加速システムの態様について説明するうえで、可能な態様のうちのいくつかについてのみ説明する。当業者には他の適用例および変形例が明らかであると思われる。したがって、本発明は実施例ほど狭義に解釈すべきではなく、むしろ添付の特許請求の範囲に従って解釈すべきである。

【００２８】

トランザクションは、本明細書では、データをある場所から別の場所に移動させる論理的な1組の段階である。場合によっては、移動させるデータは、ファイルがサーバのディスク上に存在するファイル読取りトランザクションなどのトランザクションとは無関係な送信元に存在する。他の場合には、データは、計算、参照などを求める要求に対する応答のように、送信元でトランザクション用に生成される。通常、トランザクションを開始するコンピュータ、コンピュータ装置などは「クライアント」と呼ばれ、応答するかまたは応答することが予想されるコンピュータ、コンピュータ装置などは、「サーバ」と呼ばれる。データはいずれかの方向に流れることができる。たとえば、ファイル・システム・クライアントは、ファイル読取りを要求することによってトランザクションを開始することができる。対応するデータは要求に応答してサーバから返され、したがって、この場合、データの大部分は、最初の要求の一部としてまたは後続のメッセージとしてサーバからクライアントに流れる。トランザクションは複数の部分であってよいが、簡単なトランザクションでは、クライアントが要求（明示的に要求であるかまたは要求を示すかもしくは表すデータ、メッセージ、信号など）をサーバに送信し、サーバは、クライアントへの応答（明示的に応答であるかまたは応答を示すかもしくは表すデータ、メッセージ、信号など）で応答する。より複雑なトランザクションはたとえば、サーバが要求を明確にし、クライアントの権限を検証して要求に対する応答を受信し、応答を作成するのに必要な追加的な情報を得るために必要な、ある送受信を含んでよい。

【００２９】

本明細書では、クライアントとサーバとの接続の代表的な例はパケット・ネットワークであるが、ポイント・ツー・ポイント有線チャネルやポイント・ツー・ポイント無線チャ

ネルのような他の接続手段を使用することもできる。これらの要素を、一般化し本明細書では「ノード」と呼び、ノード間の通信にチャネルを仮定する。

【００３０】

トランザクションではまず、あるノードにあるクライアントが他のノードにあるサーバに対するファイル・データを求める要求を出し、その後、要求されたファイル・データを含む応答が供給される。他のトランザクションは、ファイルの特定の部分、すべてのファイル、他のデータ構造のすべてもしくは一部であってよく、またはトランザクションを要求側から流れるデータに関係付けるか、もしくはコマンドに関係付けることができる。トランザクションの例には、「ブロック読取り」、「ファイル読取り」、「ストリーム読取り」、「このデータをブロックに書き込む」（データが要求側から流れる例）、「ファイルを開く」、「このデータに対して計算を実行する」、「これらの特性を持つeメールを得る」、「eメール送信」、「新しいeメールがあるかどうか調べる」、「ディレクトリの中身を調べる」などが含まれる。

【００３１】

トランザクションによっては、大量のデータが一方向または両方向に流れることができる。説明を明確にするために、これらの多数のトランザクション・タイプについて、あるクライアントがあるサーバに要求を出し、このサーバがクライアントの期待に応じて要求に応答する代表的な簡単なトランザクションに関して説明する。しかし、当業者なら、本開示を読んだときに、これらの概念を、クライアントとサーバとの間またはより一般的には2つのノード間の一対多トランザクションおよび多対多トランザクションに適用することができる。データ・フローを一方向において説明する場合、データが他の方向に流れることができ、かつ／または情報が一方向にしか流れないが、データおよび／もしくは信号が両方向に流れて情報の移動を実現することを理解されたい。

【００３２】

本明細書で説明するシステムのいくつかを使用すると、サーバへのクライアント・アクセス（および必要に応じてクライアントへのサーバ・アクセス）を、コンテント誘導セグメント・カット・ポイントを有する可変長セグメントのシーケンス上にマップするトランザクション・アクセルレータによって「トンネリング」することができる。各セグメントは、通常クライアントとサーバの両方の高速アクセス内の、様々な場所に記憶することができ、各セグメントはスケーリング可能な持続命名システムを用いて記憶される。各セグメントは、ファイル・システムならびに他のシステム・データ・ブロックおよび構造から結合解除することができ、したがって、複数の文脈において一致するセグメントを見つけることができる。ファイル、ブロック、または他のシステム依存構造を取り込む代わりに、セグメントを記憶し、セグメント・コンテントを表すのに用いられる参照に結合することができる。

【００３３】

図1は、このようなトランザクションを行うことのできる本発明の態様によるネットワーク化されたクライアント－サーバ・システム10のブロック図である。図1に示されているように、クライアント12は、クライアント側トランザクション・アクセルレータ（「CTA」）20およびサーバ側トランザクション・アクセルレータ（「STA」）22を介して、ネットワーク16上でサーバ14に結合されている。トランザクション・アクセルレータの位置は特定の位置ではないが、本明細書では、トランザクション・アクセルレータを、クライアント側トランザクション・アクセルレータ、サーバ側トランザクション・アクセルレータ、ピア・トランザクション・アクセルレータ、場合によってはクライアントおよびサーバによって（場合によってはさらにピアによって）使用されるトランザクション・アクセルレータを示す「TA」と呼ばれる。

【００３４】

図1には示されていないが、クライアントとサーバの間（場合によってはクライアントとクライアントの間およびサーバとサーバとの間）の追加的なパスが存在し、TAをバイパスする。このような追加的なパスを用いて、トランザクション加速によって利益を得る可

能性の低いトランザクションのような従来のトラフィックを運ぶのに用いることができる。このようなトランザクションをTAの周りにルーティングすることにより、たとえば、TAの持続セグメント記憶装置（後述）に、トランザクション加速によって利益を得る可能性の低いトランザクションのセグメントを記憶させないことによって、TAの状態をトランザクションの加速を対象とした状態のままにすることができる。

【００３５】

図示のように、CTA20は1つまたは複数のクライアントとして働くことができ、複数のCTA20をネットワーク上で実施することができる。本明細書では、特に明示しないかぎり、指標「n」は無限整数を指し、この指標を使用するたびに異なる無線整数を指すことができる。たとえば、図1は、いくつかのCTAおよびいくつかのSTAがあってよく、かつ1対1に対応する必要はないことを示している。一般に、CTAの数は、クライアントの数、予想されるクライアントの数、ネットワーク・レイアウトなどに基づいて定めることができ、一方、STAの数は、サーバの数、予想されるサーバの数、ネットワーク・レイアウトなどに基づいて定めることができる。いくつかの実現態様では、複数のサーバをサーバ・ファーム・マネージャ24に結合し、さらにSTAを介してインターネット16に結合することができる。場合によっては、クライアントは、図1の線27によって示されているように複数のCTAと対話することができ、サーバは、図1の線29によって示されているように複数のSTAと対話することができる。

【００３６】

あるCTAがあるSTAと対話し、そのCTAに接続された複数のクライアントから要求が受信された場合、対応するSTAは、各クライアント要求を、要求が対象とするサーバにルーティングする。しかし、TAは、あるクライアントからのすべてのまたはほぼすべての加速されたトランザクションがこのクライアントのSTAを通過し、あるサーバへのすべてのまたはほぼすべての加速されたトランザクションがこのサーバのSTAを通過するように、TAのクライアント／サーバにより密に結合することができる。さらに、いくつかの実現態様では、TAは状態を共用し、したがって、あるTAにおけるトランザクションは別のTAに記憶されているセグメントの利益を得ることができる。

【００３７】

クライアント接続は、従来技術のプロキシがクライアントに対して機能するのと同様に、いくつかの方法でCTAにルーティングすることができる。たとえば、ドメイン名システム（DNS）を用いた出力先変更を使用して、クライアントにサーバではなくCTAのIPアドレスを変換させ、それによって要求をCTAにルーティングすることができる。または、クライアントまたはクライアントのアプリケーションを、特定のCTAまたはアプリケーションごとの1組のCTAを使用するように静的に構成することができる。クライアント接続がCTAに到達すると、CTAは、多数の方法で働くことのできる参照プロセスを介して適切なSTAに接触することができる。たとえば、（中央の問合せ可能データベース上に維持されるかまたはCTAに組み込まれた）マッピング・テーブルを用いてCTAに適切なSTAを対象とさせることができ、また、トランザクションで転送される情報によって、CTAが、どのSTAを使用すべきかを認識することができ、また、どの転送ポートをどのSTAに中継すべきかを示す構成可能なポリシーをCTAにプログラムすることができる。同様に、STAは同様の参照プロセスを用いて、CTAから到着する新しいクライアント接続に対してどのサーバが接触すべきかを判定することができる。STAは、トランザクションにおけるデータを用いてどのサーバに接続すべきかを推定することができる（たとえば、CIFSファイル・サーバ接続を求める接続セットアップ要求と同様に、HTTP Web要求はサーバのIDを含む）。

【００３８】

図1に示されているネットワークは、今日広く使用されている各ネットワークのグローバル・インターネットワークであるインターネットであるが、代わりに他のネットワークを使用できることを理解されたい。たとえば、インターネット上のネットワーク・トラフィックは、公衆網を流れることができ、主としてTCP/IP（Transmission Control Protocol/Internet Protocol）パケット交換に基づいて生じる。しかし、本明細書に示されてい

る本発明の態様は、イントラネット、エクストラネット、仮想専用網のような公衆網ではないネットワーク上で使用することもできる。各態様は、WAN、LAN、WAN/LAN結合網、無線接続、移動リンク、衛星リンク、携帯電話網、または応答性が重要である他のネットワークと共に使用することもできる。さらに、TCP/IPは今日最も一般的なパケット交換プロトコルであり、したがってプロトコルとして好例であるが、他のネットワーク・プロトコル（イーサネットなど）を使用してもよい。上層プロトコルについては、本明細書で説明するクライアントおよびサーバ（ならびに後述のようにピア）は、HTTP、FTP、SNMP、POP3、IMAP、SMTP、NFS、CIFS、RPC、またはデータ転送用の他の公開プロトコルまたは所有権付きプロトコルを使用してよい。

【００３９】

一般的なトランザクションでは、クライアントは、ファイル、データ・ブロック図、または他のデータ単位を求める要求をサーバに送信し、サーバは、可能ならこの要求に応じたデータで応答する。たとえば、クライアントが、コンピュータ援用設計（CAD）プログラムを実行するコンピュータであり、ファイル・サーバ上に記憶されているCADファイルが必要である場合、クライアントは、このファイルを求める要求を作成し、要求をメッセージとしてカプセル化し、このメッセージをネットワーク上で適切なファイル・サーバに送信することができる。ファイル・サーバは次いで、認証を実行し、ファイル・サーバにそのファイルが存在しているかどうかを検査し、クライアントがこのファイルを有する権限を与えられておりファイルが存在する場合、ファイル・サーバは、要求されたファイルのデータを含むメッセージまたは1組のメッセージまたはパケットを作成し、これらのメッセージ／パケットを、要求を出したクライアントに送信することができる。

【００４０】

TAを使用すると、トランザクションの応答性を向上させ、すなわち、トランザクションを加速することができる。代表的な環境では、クライアントとCTAの間のリンク27は、ローカル・エリア・ネットワーク（LAN）リンクなどの高速リンクであり、ネットワーク16上のリンクは、呼出し時間および帯域幅に関してより低速である。「呼出し時間」は、メッセージが送信されてから受信されるまでの時間（通常、時間単位で測定される）を指し、「帯域幅」は、特定のタスクについてリンク上でどれだけの容量（通常、単位時間当たりビット数で測定される）を転送できるかを指す。多くの場合、帯域幅が狭いと呼出し時間が長くなる可能性があるが、これらの因子は、帯域幅が広くそれにもかかわらず呼出し時間が長くなるように互いに独立していてよい。リンクの信頼性や帯域幅の使用法のような他の因子が応答性および／または帯域幅コストに影響を与えることがある。

【００４１】

TAを用いた代表的な要求トランザクションでは、クライアント12は、要求メッセージを送信することによってサーバ14とのトランザクションを開始する。上述のように、トランザクションで使用されるビット数が少ないか、または他の因子が存在する場合、TAを使用してもトランザクションが加速されず、したがって、トランザクションが従来のパケット・パスを利用することがある。しかし、トランザクションはいずれにしても、後述のように有用なTAを通過することができ、したがって、TAはトラフィックのより完全なビューを有する。一例として、クライアント要求がCTAを通過する場合、CTAは要求を記憶し、応答を要求に一致させ、クライアントに追加的なサービスを提供することができる。CTAは、要求を用いて、将来のイベントがどのようなものであるかを推定し、この推定を用いて、トランザクション加速プロセスをさらに最適化することもできる。

【００４２】

サーバ14は、要求を受信すると、要求に対する応答を作成し、サーバ14に結合されているSTA22を介してクライアントに向けて送信する。基本的な実現態様では、各クライアントは1つのCTAに結合され、各サーバは1つのSTAに結合されるが、より複雑な実現態様では、サーバは、複数のSTAに結合することができ、かつ何らかの最適化論理を用いてどのSTAをいつ使用すべきかを判定することができる。クライアントは、複数のCTAに結合することができ、かつ何らかの最適化論理を用いてどのCTAをいつ使用すべきかを判定すること

ができる。

【００４３】

CTA20は、要求を変更せずに適切なSTA22に送信することができ、かつ／または受信側STA22は、応答をサーバから受信し、変更せずに適切なCTA20に送信することができる。しかし、要求または応答が大量のデータを含んでいる場合、受信側にデータのセグメントを記憶し、送信側で、記憶されたセグメントの参照でデータを置き換えることによって、本明細書で説明するようにデータを圧縮する場合には、このような例では顕著なトランザクション加速が予想される。場合によっては、このような置換ではトランザクションが加速されず、それにもかかわらずデータの「ポンプへの装填」のような利益を得ることができ、したがって、受信側は、後でこれらのセグメントを参照する送信されたデータを再構成する際に使用できるセグメント・データを有する。このような概念について図2を参照してより明確に説明する。

【００４４】

図2およびその他の図を詳しく見ると分かるように、トランザクション要求および応答は、クライアントからサーバに直接送られるのではなくTAを通してルーティングされる。もちろん、構成によっては、CTAとクライアントおよび／またはSTAとサーバは、明白な再ルーティングが必要になるように密に統合される。それにもかかわらず、データがルーティングされると仮定すると有用である。なぜなら、少なくとも、クライアントからのトラフィックをCTAを通してルーティングすることができ、サーバからのトラフィックをSTAを通してルーティングすることができるが、トラフィックがTAをバイパスすることもできることが明確になるからである。TAは、トランスポート・ネットワーク・レベルで動作できるため、トランザクションを作業の単位として処理することができる。

【００４５】

加速すべきトラフィックを容易にルーティングする1つの構成では、接続プロキシが使用される。したがって、CTAは、クライアントがトランザクションを行うサーバの接続プロキシとして働き、STAは、サーバが応答するクライアントの接続プロキシとして働く。たとえば、場合によっては、CTAがクライアントからの新しいトランザクションに出会うがSTAからのトランザクションには出会わないようにセットアップされ、STAがサーバからの新しいトランザクションに出会わず、CTAからのトランザクションに出会うようにセットアップされることを除いて、CTAとSTAが実質的に同様に構成される対称TAによって、TAシステムを実現できることを理解されたい。

【００４６】

図2は、CTA20、STA22、およびそれらの相互接続を詳しく示す、システム10の一部のブロック図である。1つのクライアントおよび1つのサーバしか示されていないが、図1の様々な要素についても、図示されていない場合でも存在してよいことを理解されたい。たとえば、CTA20は、複数のクライアントからのトランザクションを取り扱うことができ、STA22は、複数のサーバからのトランザクションを取り扱うことができる。図2に示されているように、クライアント12はCTA20のクライアント・プロキシ30に結合されている。クライアントとの間の多重化トラフィックおよび逆多重化トラフィックの他の形態を使用してよいが、この例では、複数のクライアントからCTA20のデータを受信し、CTA20のデータを複数のクライアントに送信するのにクライアント・プロキシが用いられている。図2に示されているCTA20の他の要素には、トランザクション変換器（TT）32、逆トランザクション変換器（TT^{-1}）34、持続セグメント・ストア（PSS）36、および参照リゾルバ（RR）38が含まれる。サーバ14は、トランザクション変換器（TT）42、逆トランザクション変換器（TT^{-1}）44、持続セグメント・ストア（PSS）46、参照リゾルバ（RR）48のようなCTA20の要素と同様の要素を含むSTA22のサーバ・プロキシ40に結合されている。

【００４７】

クライアント12は、TT32およびTT^{-1}34に結合されたクライアント・プロキシ30に結合されている。TT32は、PSS36およびCTA20とSTA22の間のネットワークに結合されている。TT^{-1}34は、PSS36、クライアント・プロキシ30、RR38、およびCTA20とSTA22との間のネットワ

ークに結合されている。図示のRR38は、PSS36およびCTA20とSTA22との間のネットワークにも結合されている。

【００４８】

図の反対側で、サーバ14は、TT42およびTT^{-1}44に結合されたサーバ・プロキシ40に結合されている。TT42は、PSS46およびSTA22とCTA20との間のネットワークに結合されている。TT^{-1}44は、PSS46、サーバ・プロキシ40、RR48、およびSTA22とCTA20との間のネットワークに結合されている。図示のRR48は、PSS46およびSTA22とCTA20との間のネットワークにも結合されている。

【００４９】

CTA20および／またはSTA22のいくつかまたはすべての要素を、各要素間の明白な接続が存在しないが、それにもかかわらず論理結合が存在するように、CTA20またはSTA22に組み込むことができることを理解されたい。たとえば、CTA20全体を、データ・メモリと、クライアント・プロキシ、TT、TT^{-1}、およびRRを実現するための命令を、そのような命令がプロセッサによって実行されるときに含むプログラム・メモリと、プロセッサとを有する単一のプログラムとして実現することができる。そのような実現態様では、データ・メモリを、プロセッサが命令を実行するのに必要な変数、クライアント・プロキシ、TT、TT^{-1}、およびRRの状態、ならびにPSSの内容を保持するように論理的に区画することができる。

【００５０】

PSSは、ディスク・サブシステム、メモリ・サブシステム、またはその一部であってよい。PSSは、ディスク・バッキング・ストア、データベース・サーバ、データベースなどを有するメモリ・サブシステムであってもよい。

【００５１】

図示の接続のうちで、矢印は、情報の流れの最も一般的な方向を示しているが、情報は他の方向に流れることができ、単一の方向における情報の流れは、逆方向に流れるデータも含んでよい。たとえば、TT32は概して、TT^{-1}44の方向に情報を送信するが、確認、ハンドシェークなどのデータはTT^{-1}44からTT32に流れることができる。

【００５２】

いくつかの接続は、CTA20とSTA22との間（たとえば、TTとTT^{-1}とRRとの間）に延びる点線として示されている。点線は別々の線として示されているが、これらの線がそれぞれの異なるネットワーク接続を表すことができ、すなわち、別々のパケットが共通のネットワーク接続上を流れるか、場合によっては共用パケットが図示の論理接続間を流れることを理解されたい。したがって、点線の接続は、複数のポート番号および／または複数のIPアドレスを含む互いに独立した接続であってよいが、共通のポート番号および共通のIPアドレスを用いて共通のパスを介することなどによる1つのパケット交換接続上の3つの論理接続であってもよい。

【００５３】

クライアントとCTAとサーバとSTAとの間の点線でない線は、これらの接続が、「インターネット/WAN/など」と示されているTA同士の間の接続よりも高い性能を有する可能性が高いことを示すように「LAN/直接」と示されている。前者の例には、LAN、ケーブル、マザーボード、CPUバスなどが含まれる。システムは、TA同士の間の接続の方がより性能の高い接続である場合でも動作可能であるが、トランザクション加速の利点のいくつかが得られなくなる可能性がある。

【００５４】

動作時には、CTAおよびSTAは、そのトランザクションのペイロードを調べ、トランザクションと無関係であってよい固有の命名方式を用いて、これらのペイロードから得られるデータのストリングまたはその他のシーケンス（「セグメント」）を記憶／キャッシングする。ペイロードをあるTAから別のTAに送信する際、TAは、セグメント・データをそのセグメント・データの参照で置き換えることができる。この置換えを行う必要があることが示されるのは、たとえば、セグメント・データが、その一意に命名されたセグメント・デ

ータが前のトランザクションに現れるか、または他のプロセスによって受信側に送信されているため、受信側がこのセグメント・データを有することを送信側が予想できるようなセグメント・データであるときであるが、セグメント・データを参照で置き換えるか否かを判定する場合にその必要性を示す他のことを使用してよく、また、そのようなことをまったく使用しなくてもよい。場合によっては、関連するデータの量が少ない場合のような、加速が予期されない場合には、セグメント化および置換えは行われない。トランザクションのセグメント化される部分は、トランザクションが、受信側において、再構成するのに十分な程度に識別可能であるかぎり、送信されるデータの任意の部分であってよい。

【００５５】

各セグメントを一意に命名することができ、かつ名前はトランザクションと無関係であってよいので、あるトランザクションに現れるセグメントを両方のTAに記憶し、他のトランザクションの加速に使用することができる。たとえば、クライアントがいくつかのファイル要求トランザクションを開始する場合、各ファイルが共通のデータを有する場合には、その共通のデータをセグメントとして形成することができ、第1のそのようなセグメントが送信された後、その共通のデータを含むファイルを求める他のすべての要求は、再構成されたファイルを要求を出したクライアントに送信する前にCTAによって置き換えられる、その共通のデータに置き換わるセグメント参照を有する。同様に、1つのクライアントが複数のクライアントを取り扱う場合、1つのクライアントのセグメントを他のクライアントに使用することができる。

【００５６】

トランザクションがファイル・トランザクション以外である場合、同様の加速が可能である。たとえば、CTAがeメール・クライアントに結合され、STAがeメール・サーバに結合されている場合、多数のクライアントがCTAを介して要求しているeメール添付ファイルを、CTAがその添付ファイルの内容を得た後でセグメントとして表すことができ、その後クライアントがその添付ファイルを要求するたびに、応答側のSTAは、添付ファイルをセグメント参照で置き換え、受信側のSTAは参照を記憶されている添付ファイルで置き換える。添付ファイルがトランザクションとは無関係なセグメントとして記憶されるので、同じセグメント・データがファイル・トランザクション、他のeメール・トランザクション、またはその他のトランザクションに存在してよく、それぞれの場合に、送信側はデータをセグメント参照で置き換え、受信側はセグメント参照をセグメント・データで置き換える。

【００５７】

このような手法にはいくつかの利点があることに留意されたい。トランザクションの結果がキャッシングされ、そのトランザクションが繰り返されたときに再使用され、キャッシュが無効化されないキャッシングとは異なり、セグメントをいくつかの互いに無関係のトランザクションで使用することができ、セグメントを任意のカット・ポイントで分離する必要がない。セグメント名およびコンテントは、任意の特定のビット・ストリームやトランザクションと無関係であってよいので、システム構成要素がクラッシュしてリブートしたり、新しい構成要素を追加したり、セグメント・ストアを消去したりする場合でも、任意の時間にわたって持続記憶装置に残ることができる。

【００５８】

受信側は、セグメント・データを得て、その持続ストアに含め、かつ／または送信側からの参照のシーケンスの伝送を受信する前、受信中、または受信した後に復号化することができる。好ましくは、セグメント・データは、可能ならトランザクションの応答性を向上させるように得られる。たとえば、セグメントが必要であることを予想できる場合、セグメント・データを、それが必要になったときにより高速に得られるように、必要になる前に送信しておくことができる。しかし、受信側のTAが、記憶されているセグメントを有さず、トランザクション中にすべてのセグメントを得る必要がある場合のような、いくつかの場合には、送信する必要のあるデータの総量が減らないためトランザクション加速が行われない可能性がある。

【００５９】
要求がTAを通って流れると仮定すると、クライアント12は要求をクライアント・プロキシ30に送信し、クライアント・プロキシ30は要求をTT32に送信し、要求を修正するか、または単に転送する。TT32は、どのようにして要求を変換すべきかを判定し、必要に応じてセグメントおよびその参照をPSS36に記憶し（以下に詳しく説明する）、変換後の要求または未修正の要求をTT^{-1}44に送信し、TT^{-1}44は、必要に応じて逆変換を実行し（以下に詳しく説明する）、要求をサーバ・プロキシ40に送信し、さらにサーバ14に送信する。応答にも同様のパスが使用される。
【００６０】
メッセージ（クライアント要求メッセージやサーバ応答メッセージなど）が変換された後、逆トランザクション変換器はそのPSSのコンテントを用いてメッセージを再構成する。単純な場合には、送信側（クライアントまたはサーバ）用のTTが、メッセージのセグメントを識別し、識別されたセグメントを参照で置き換え、参照－セグメント対をPSSに記憶することによってメッセージを変換する。コンテントに基づいてデータをインテリジェントにセグメント化するいくつかの技術がMcCanne IIに記載されている。セグメント・データの代わりに参照を送信することによって、トランザクション中のTA同士の間の総トラフィックが少なくなり、またはおそらく、トラフィックの大部分がより重要でない時間またはより重要でないパスに移動させられる。
【００６１】
受信側のTAが、送信側のTTによって使用される参照－セグメント対をそのPSSに有している場合、受信側のTT^{-1}は、参照をそれに対応するセグメント・データで置き換えることによって、送信されたデータを再生することができる。受信側TAは、そのPSSに記憶されるセグメント・データをサイド・チャネルから得るか、または送信側TAからのトラフィックの一部として得ることができる。したがって、送信側TAから受信側TAに送信されるデータは、セグメントの参照と、セグメント・データの参照からのマッピングを表す「バインディング」との両方を含んでよい。もちろん、毎回セグメントが参照で置き換えられ、参照とバインディングの両方が送信される場合、節約される帯域幅は少なく、実際には帯域幅は増大する。しかし、送信側は、受信側がすでにバインディングを有していると考えられる場合、バインディングを省略することができ、それによってトラフィックは実質的に低減する。このプロセスの利益を得るうえで、受信側が何を有しているかについて厳密に知る必要はないことに留意されたい。
【００６２】
場合によっては、要求を満たすのに必要なすべてのデータがクライアントのPSSに存在し、したがって、代わりにPSSをキャッシュとして用いてキャッシング方式を使用した場合、サーバにメッセージを送信する必要はなくなる。しかし、この場合、CTAは、クライアント・トランザクションを理解し、かつPSSに存在するデータから適切な応答を作成するのに十分な程度にインテリジェントである必要がある。これは、特に多数の異なるアプリケーション・タイプおよびクライアントとサーバの関係がCTAによって代理され、クライアントとサーバのある対話が単に、キャッシングを受け付けない（たとえば、ファイル・システム書込み動作、データベース更新トランザクション、ファイル削除など）とき、一般に難しい問題である。したがって、キャッシング方式の矛盾を回避できるように、本明細書に記載されたTAを使用し、メッセージを何らかの方法でサーバに送信することが好ましい。たとえば、ファイル・サーバがファイルを求めるすべての要求を受信した場合、ファイル・コンテント全体がクライアントに存在する場合でも、サーバは各要求を追跡することができ、サーバは、実質的なファイル・データをネットワークを介して送信する必要がないにもかかわらず、ファイル・ロッキング・プロトコルのような複雑な動作を実現することができる。
【００６３】
TAシステムの好ましい態様では、上述の利点が自動的にもたらされる。たとえば、STAは、各トランザクション・ペイロードをセグメント化し、各セグメントを参照で置き換え

る。STAが、CTAが有していると考えるセグメント・データについては、STAは、CTAが有することが分かっている参照をそのセグメントに使用する。サーバでデータが変更されると、STAは、PSS内の既存のセグメントの修正を試みるのではなく、変更されたデータを表す新しいセグメントを作成し、CTAはこのセグメントを有していないと仮定することができる。この場合、STAは、変更されたデータを表す新しいセグメントの新しい参照を使用する。受信側TA（この例ではCTA）では、古いデータの参照を受信側のPSSに記憶されているバインディングから得ることができるが、新しい変更されたセグメントについては、参照は、送信側からのストリームに含まれるバインディングから得られる。それらのバインディングは、受信側によって後に変換するため利用可能であるように、受信側TT^{-1}によって受信側PSSに記憶される。

【００６４】

さらに、参照は(後述のように)グローバルに一意であるため、この例の説明とは異なりSTAとCTAの対だけでなく、ネットワーク内の任意のTAによって使用することができる。たとえば、CTAは、異なるSTAと通信し、前述のSTAによって割り当てられた参照を使用することができる。2つのSTAは、将来通信する場合に、両方の装置に伝播されたセグメント・バインディングの利点を直ちに享受することになる。

【００６５】

いくつかの方式は、命名された各セグメントが任意の所与の時間にシステム全体にわたって一意の名前を有するように使用可能である（すなわち、異なるデータを有する2つのセグメントに誤って同じ名前が割り当てられることがなくなる）。1つの手法では、あらゆるセグメント参照が大きな乱数として生成され、この場合、一意の参照の数は、すべての可能な大きな乱数のスペースよりもずっと少ない。この方式は、2つのセグメントが同じセグメント参照を有するが異なるセグメント・データを有し、それによって、受信側TAが誤ったデータを含むメッセージを再構成するわずかな可能性が存在するため、それほど望ましくない。

【００６６】

他の方式では、各セグメント参照がセグメント・データのハッシュであり、非常にまれな場合を除いて、セグメント・データが異なればハッシュも異なるように、セグメント・データからハッシュが生成される。この場合も、このまれな場合は、同じ参照を有するが異なるセグメント・データを持つ2つの変質セグメントがシステムに存在するかぎり、常に問題がある。乱数の場合と異なり、この問題は、データ・ストリームに特定のデータ・パターンが存在するたびに起こる。

【００６７】

上記の問題を解消する1つの簡単な手法は、各送信側TAが一意のID（ネットワーク全体にわたってグローバルに一意のIPアドレスが使用されるときのホストIPアドレスや、ホストMACアドレスや、割り当てられる一意の識別子や、その他の手段など）と通し番号の組合せからセグメント参照を生成する手法である。たいていの実現態様では、一意の通し番号の最大数が制限され、したがって、一意の通し番号を最終的に再使用する必要がある。しかし、数100万年にわたって供給し続けることのできる十分に大きなラベル番号スペースを使用することによって、名前スペースを実際上、制限されないようにすることができ、特殊な取扱いが不要になる。大きなラベルは、ラベルのフットプリントを小さくすることができるように圧縮することができる。

【００６８】

ラベルが順次割り当てられ、対応するセグメントが同じシーケンスに現れることが多いので、（ネットワーク全体にわたってだけでなく、システム全体にわたって必ず使用されるラベルのストリングを表すデータ構造でも）実際にはラベルの非常に良好な圧縮を行うことができる。送信側TAの出力ストリーム上で追加的な圧縮も可能である。たとえば、受信側TAが送信側TAを識別することができ、送信側TAの参照が送信側TAの一意のIDを含む場合、そのIDが、送信されるデータに現れる必要はない。これは、受信側TAが参照を形成するうえでどんなIDをすべきかが分かっているからである（ただし、一般に、送信側TAがそ

れ自体のバインディングだけでなく他のTAから送信されたバインディングおよびラベルも参照するときには余分な情報を伝達しなければならない）。この手法の他の1つの利点は、TAがセグメント参照のID構成要素から、PSS用の各セグメントの送信元を識別し、統計分析、診断などで使用できることである。

【００６９】

ラベルがシステムの予想寿命の間に再使用されるようになっているシステムでは、システムは好ましくは、参照バインディングを「満了」させる機構を含み、この満了はネットワーク内のすべてのTAに伝搬される。ある手法では、各セグメントが、システム内の、そのセグメントを使用する各構成要素によって容易に推定できる一定の寿命を有するように、各セグメントにタイムスタンプが付けられる。タイムスタンプを各ラベルに粗に割り当てる（たとえば、タイムスタンプが1日に一度のみ変更される）場合、ラベル圧縮によって、タイムスタンプの割当ておよび通信に関連するプロトコル・ヘッダの大部分が不要になる。それによって、TAは特定の1組のラベルを再使用しても安全なのはいつかを推定することができる。

【００７０】

セグメント名スペースを管理する他の方法では、一意の参照を集中的に割当てる。このような場合、送信側TAは、参照が一意であることを保証する送信元に参照または参照のブロックを要求する。さらに、参照または参照のブロック図の割当てを暗黙的に再使用できるように、各割当てに最高寿命が割り当てられる。

【００７１】

あるバインディングが受信側TAに実際には存在しないのに存在すると送信側TAが仮定する場合がある。これは、受信側TAにPSSオーバフロー、破壊、電力損失などが存在するか、または受信側TAが故意にバインディングを削除した場合に起こる可能性がある。このような場合、受信側TAは、トランザクションを中止したり、トランザクションを失敗として報告する必要なしにセグメント・データを得ることができる。これによって、システムは、ディスクが満杯になったり、ディスクに障害が起こったり、ネットワークに障害が起こったり、システム・クラッシュが起こったりしたためにデータが失われた場合にうまく対処することができる。受信側TAがバインディングを有していると送信側TAが仮定する場合、送信側TAは、そのバインディングの参照を用いてメッセージを参照するが、バインディングのセグメントを含めない。受信側TT^{-1}がこの参照を変換しようとすると失敗する。この場合、受信側TT^{-1}は、そのRRに変換要求を送信し、次いでRRは送信側のRRに要求を出す。TT^{-1}は、単に遮断し、おそらく、データが利用可能であることを示すイベント・トリガのために、必要なデータが受信されたときに再始動することができる。このプロセスはTT^{-1}に対して透過的であってよい（応答を得る際の遅延を除く）。セグメント・データが受信側で受信された後、受信側のRRは受信側のTT^{-1}にデータを供給するか、または単に受信側のPSSに格納することができ、後者の場合、受信側のTT^{-1}からセグメント・データにアクセスすることができる。送信側のTTは、そのPSSに必要に応じてバインディングを追加する際、このバインディングを保証された最短時間にわたって維持する。そのため、送信側のTTがセグメント・データを参照で置き換える際、受信側のRRが送信側のRRにそのセグメント・データを求める要求を出すときに、送信側におけるセグメントの保証された「寿命」が受信側がセグメント要求を出すのに必要とする最長時間よりも長いかぎり、セグメント・データが送信側のPSSに存在することを保証することができる。

【００７２】

図3は、簡単なPSSのバインディング・テーブルのデータ構成の図を含んでいる。図3に示されているように、バインディング・テーブルは、(R_1,S_1)、(R_2,S_2)のような複数のバインディングを含み、この場合、R_iはi番目のバインディングの参照ラベルであり、S_iは、i番目のバインディングのセグメント・データである。各バインディングのタイムスタンプをバインディングの寿命を示すのに用いることができる。バインディング・レコードは、表1にリストされているようなフィールドおよび／または同様のもしくは他のフィールドのような、図3には示されていない他のフィールドと、場合によっては他のテーブ

ル、データ構造、オブジェクト、および／またはコードとを含んでよい。
【００７３】
（表１）
－アクセス回数
－最後のアクセス時間
－最後の修正時間
－寿命
－符号方法識別子（たとえば、未符号化生データ、ランレングス符号化、MD5符号化、暗号化）
－フィンガープリント
－誤り補正データ（セグメント・データが存在しない場合）
－バインディングを作成した送信側の表示（バインディング「所有者」）
－作成時間（セグメントをタイムアウトさせるうえで有用、寿命フィールドを使用することなど）
－他のフィールド
【００７４】
他のいくつかのデータ構造には、PSSのコンテントを探索するかまたは他の方法で処理するための参照の索引、他のフィールドの索引、セグメントの索引などを含めてよい。セグメントは、符号化プロセスに有用であってよい多数の方法で索引付けすることができるが、一態様は、セグメントを含むすべてのデータにわたって算出される公知のハッシュがキーとして用いられるセグメントの索引を作成する。符号化方法識別子を使用する場合、セグメント・データを誤り補正、暗号化などのために符号化することができる。
【００７５】
あるセグメント・データでは、セグメント・データを圧縮してPSSに必要な記憶量を少なくし、バインディングを設定するのに必要な伝送オーバヘッドを低減させるのが適切である。たとえば、送信側TAは、逐語的セグメント（文字通りトランザクション・データのサブストリングまたはサブシーケンスを表す）を送信しキャッシングするのでなく、セグメントの反転可能な機能を送信し、たとえば、セグメントの誤り補正符号化済みブロック、セグメントの暗号、セグメントの署名などを転送することができる。これによって、受信側TAは、適切に符号化されたデータの共通の集合から様々なセグメントを復号化することができ、符号化後のデータのある部分が失われるかまたは破壊された場合、それにもかかわらず元のセグメントを再構成し、したがって、クライアントまたはサーバにおける変更を必要とせずにリンクに誤り補正を付加することができる。
【００７６】
どのセグメントをどの受信側が知っているかを追跡するための他のフィールドがPSSに存在してよい。いくつかの実現態様では、送信側は単に、データをセグメント化し、受信側が結果に対して何がするかとは無関係な参照を作成するが、他の実現態様では、送信側は、受信側が、どの受信側がどのセグメントをすでに受信しているかを追跡することなどによって特定のバインディングを有するかどうかを判定するのに用いることのできない情報を維持する。このような情報の記憶は、どの受信側がどのセグメントを有するかをブルーム・フィルタ（すなわち、送信先のハッシュによって、まれに擬似正値を与えることがあるが擬似負値を与えることのないベクトルとして索引付けされたビット・ベクトル）に記録することによって最適化することができる。
【００７７】
いくつかの実現態様は、新しいセグメントのバインディングを自動的に得るのが第1のクライアント・プロキシだけであるため、サーバ・プロキシが、新しいエントリを作成するときにのみセグメント・バインディングを含み、かつセグメントを必要とする他のクライアント・プロキシがそれを要求することが必要になるようにヒューリスティックを使用することができる。
【００７８】

TAは、クライアントがファイルを閉じるときに特定のサーバ上の特定のファイルに関するクライアント側PSS内のすべてのセグメントを削除するように指示するヒューリスティックなどのPSSホウスクリーニング用のルーチンを含んでよい。サーバ側PSSも、対応するセグメントを削除するか、またはすべてのクライアントがファイルを閉じるまでこれらのセグメントのハウスクリーニングを延期することができる。他のハウスクリーニングでは、寿命を超えたか、またはしばらく使用されていないセグメント・エントリを削除してよい。他のヒューリスティックは、特定のセグメント・バインディングをいつ使用し破棄すべきかを示すことができる。

【００７９】

PSSの構成はいくつかの利点を有し、そのうちのいくつかは本開示を読んだときに明らかになろう。セグメント化を様々なカット・ポイントで行うことができ、かつセグメントはトランザクションと無関係であってよいので、各セグメントはPSS内に任意の期間にわたって存在し、セグメントが作成され記憶されたトランザクションとはまったく無関係なトランザクションに使用することができる。セグメント参照は一意のセグメント・データに対して一意であるので、受信側はセグメント参照用のセグメント・データを常に正しく識別することができる（受信側がセグメントを有する場合）。これは単に結果をキャッシングするのよりも優れている。この場合、適応的コードブックのバインディングなど、局部信号統計による圧縮も改善される。システム構成要素がクラッシュしてリブートし、新しい構成要素が付加され、持続セグメント・ストアが消去された場合でも、セグメント名およびコンテントは任意の特定のビット・ストリームとは無関係である。PSSが、各セグメントが永久的に記憶され、したがってパージできないことを意味するのでなく、少なくともいくつかのセグメントが少なくとも1回のトランザクション以上持続することを意味することを表すのに「持続」が用いられていることを理解されたい。

【００８０】

図4は、エンコーダ140および142を示している。TA用のTTはエンコーダ140のみであってよいが、TTは他の機能または要素を含んでもよい。図示のように、エンコーダ140は、符号化すべきデータ用の入力と、入力データに関する制御パラメータおよび帯域外情報を符号化するための制御入力とを有している。エンコーダ140は、PSS 142に記憶される符号化済みデータおよびセグメント・バインディング用の出力を有するように示されている。動作時には、エンコーダ140は、入力データを処理し、データのセグメントを識別し、セグメントのデータを参照で置き換え、セグメント・データおよびセグメント参照をバインディングの形でPSS 142に与え、符号化済みデータを出力する。図4に示されているように、結果として得られる符号化済みデータは、参照、バインディング、および残留データ（参照で効率的に表すことのできないデータなど）を含んでよい。本明細書では、残留データを「参照無しセグメント」とも呼ぶ。いくつかの態様では、セグメント化されたが参照されていないデータとセグメント化されていないデータとの間に違いが存在する。前者には、セグメントの明確な開始位置および終了位置があるが、セグメント・コンテントがセグメント参照で置き換えられることはなく、後者では、セグメントの開始位置も終了位置もないなどの違いがある。以下の説明を簡単にするために、この違いを無視する。

【００８１】

エンコーダ140の他の出力は、着信データを復号化する際に使用できる（または要求に応じて他のTAに供給される）PSS 142用のセグメント・バインディングである。エンコーダ140の制御入力はターゲット・セグメント・サイズを含んでよく、帯域外情報は、セグメントのデフォルト寿命、データ供給源に関する情報などを示すパラメータを含んでよい。ターゲット・セグメント・サイズは、セグメント化プロセスによって生成されるセグメントの平均サイズを調節するパラメータである。一般に、セグメントの長さは様々であり、そのサイズがある分布を有し、ターゲット・セグメント・サイズは、セグメント化プロセスによって生成される平均的なそのようなサイズを調節する。セグメント・サイズは一定にすることができるが、セグメント・サイズを変化させることができ、したがって、各セグメントは、システムが対処するデータが任意の一定のセグメントにセグメント化され

る場合よりも頻繁に整合する。

【００８２】

TTは、作成したバインディングを復号化に使用できるようにそれ自体のPSSに記憶すると共に、バインディングの「所有者」（すなわち、バインディングを作成したTA）がそのバインディングを追跡し、他のTTに供給し、かつ後でデータが符号化されるときに参照することができるように（セグメント・データが繰り返される場合にセグメント参照を再使用できるように）PSSに記憶する。

【００８３】

バインディングの所有者のTT^{-1}は、セグメント・データのシーケンスが往復し、すなわち、STAからCTAに流れるか、または逆にCTAからSTAに流れるときのように、このようなバインディングを再使用することが多い。これは、たとえば、ユーザがファイルを編集する場合に起こることがある。ユーザのファイル・クライアントがファイル・データを要求し、サーバがファイルを供給し、ユーザがファイルを編集する間、ファイル・データのバインディングがCTAのPSSとSTAのPSSの両方に存在する。ユーザがファイル・データを書き戻す場合、変更されなかった部分は、ファイル・データが最初にユーザのクライアントに送信されたときに作成された参照ラベルによって完全に表すことができる。この場合、CTAは、データをSTAに送り返す際に新しいバインディングを作成するのではなく、単に、データをSTAに送り返すときに新しいバインディングを参照する。他の例には、クライアントは（IMAPやPOPのようなあるプロトコルを介して）eメールを要求し、次いでそれをネットワーク上で（SMTPのような他のプロトコルを介して）送り返すeメールが含まれる。この場合、STAのTT^{-1}は、SMTPトランザクションとIMAPトランザクションまたはPOPトランザクションとの両方がSTA/CTA対を通って流れると仮定して、eメールが最初にクライアントに送信されたときにSTAのTTによって作成されたバインディングを使用することができる。他の例では、ユーザが、HTTPトランザクションとCIFSトランザクションの両方がSTA/CTA対を通って流れると仮定して、情報を（HTTPを介して）WebサイトからCIFSを介してファイル・システムにコピーする。

【００８４】

PSSのこの特性のために、クライアントとサーバは、帯域幅をほとんど使用せず、かつクライアントやサーバを変更することなく、大きなデータ・ブロックを効果的に送受信することができる。これは、2人以上のユーザが大きなCADファイル上で共同する場合など、大きなファイルが移動させられわずかに変更される場合に特に有用である。本明細書に示すシステムを用いると、ネットワーク性能を、ユーザがさらに、リモート・アクセス、ファイルのローカル・コピーの記憶、ファイルの読取り専用コピーのプッシュ・アウトのようなネットワーク・ボトルネックに取り組まなくても済むようにするのに十分な性能にすることができる。

【００８５】

入力データをコンテントに従ってセグメント化する場合、ビット・シーケンスがどこで発生したかとは無関係に、同じビット・シーケンスから同じセグメントが得られる可能性が高い。このことは、繰り返されるビット・シーケンスが効果的に認識され参照されるので有利である。しかし、性能の向上が強く要求されている場合、外部因子が有効になることがある。たとえば、欠点を補うだけの利点がある場合に、場合によっては、1つのビット・シーケンスに対して複数のセグメントが作成されることになる、トランザクションに関するいくつかのパラメータを使用することができる。ある手法では、外部因子は、PSSにどんなセグメントが存在するかであり、セグメント境界は、すでにPSSに存在するのはどんなセグメントかに基づいて決定される。これは上述のより基本的な手法ほどスケーリング可能ではないが、セグメントの再利用を改善し、したがって、いくつかの利点がある。

【００８６】

これを一例として示すことができる。ペイロードが通常一方向にカットされるが、異なる1組のカットがPSSにすでに存在するセグメントにほぼ一致する場合、より大きな圧縮が

実現される。しかし、送信側は、利得を維持できるように、受信側TAが有する可能性が高いのはどのセグメントかについてのある考えを有する必要があり、したがって、送信側TAは、送信側の大部分のPSSセグメントが受信側のPSSに存在しないことが分かっている場合には、送信側のPSSに基づくカットを行わない。

【００８７】

図5は、デコーダ150およびPSS 152を示している。TAのTT^{-1}はデコーダ150のみであってよいが、TT^{-1}は他の機能または要素を含んでもよい。デコーダ150は、図4に示されているデコーダ140によって出力されたであろう符号化済みデータを受信する。デコーダ150は、それが受信したデータ内のバインディングに出会うと、そのバインディング内のセグメント・データを用いて元のデータを再構成することができ、かつバインディングをそのPSSに記憶することもできる。デコーダ150は、バインディングのない参照に出会うと、その参照を用いてPSS 152からセグメント・データを得てセグメントを再構成することができる。PSS 152内にセグメント参照が見つからない場合、デコーダ152は、そのセグメント・データを求める要求を送信することができる。

【００８８】

図6は、入力データがセグメント化され、データ・セグメントの参照によって表される符号化プロセスの図である。図6に示されているように、生入力データがバッファ160にロードされる（ただし、これは必要に応じてバッファリングなしで行うことができる）。次いで、生入力データはセグメントに分割される。各セグメントをそれに隣接する近傍から分離する「カット・ライン」をどこに定めるかを決定するいくつかの技術を利用することができる。セグメント化のいくつかの手法がMcCanne IIに記載されている。使用できる他の手法は、カット・ラインを一定の間隔で配置するか、または行の終わりマークのような生入力データに存在する一定のデータ・シーケンスに対して配置する簡単な手法である。ただし、このような手法では、最適なセグメント化方式が得られないことがある。

【００８９】

しかし、カット・ラインが決定され、図6の例では、バッファ160内の生入力データがセグメントS_A、S_B、S_C、S_D、S_E、およびS_Fに分割される。この例では、最初の5つのセグメントが参照で置き換えられ、参照はR_{15}、R_{16}、R_{17}、R_3、およびR_8になる。参照が必ずしも順序正しくなく、かつこの参照（たとえば、R_3やR_8）がすでに出会ったセグメント・データに対してなされることがあり、その場合、新しいセグメントは使用されず、参照が既存のセグメントに対してなされることに留意されたい。あるセグメント（たとえば、S_F）は参照で置き換える必要がないことも図示されている。

【００９０】

生入力データは、生入力データから生成することのできる出力データおよびバインディングによって完全に表すことができる。バインディングは、そのバインディングを生成したTAのPSSと、他のPSSとに与えられ、バインディングのいくつかまたはすべてを出力データの一部として送信することができる。この例では、新しいバインディングは(R_{15}, S_A)、(R_{16}, S_B)、および(R_{17}, S_C)である。この例では、バインディング(R_3, S_D)および(R_8, S_E)は必要とされない。というのは、セグメントS_DおよびS_Eのセグメント・データは既知であり、R_3およびR_8の参照と一緒に記憶されているからである。

【００９１】

図7は、図4のエンコーダによって出力し図5のデコーダによって復号化することのできるデータを復号化するプロセスを示すフローチャートである。このプロセスの各段階を「S1」、「S2」などと呼び、各段階は一般に、特に明示しないかぎり順序正しく進行する。第1の段階（S1）で、参照付きデータ（たとえば、参照で符号化されたデータ）を受信しトークンとして解析する。トークンを検査し（S2）、そのトークンが参照でない場合は参照無しセグメントである可能性が高く、したがって、トークンを直接出力する（S3）。しかし、トークンが参照である場合、デコーダは、参照がデコーダをサポートするPSSに存在するかどうかを検査する。存在する場合、デコーダはPSSからその参照付きセグメントを取り出す（S5）。存在しない場合、デコーダは、デコーダをサポートする参照リゾツバ

に変換要求（S6）を送信し、変換された参照付きセグメントを参照リゾルバから受信する（S7）。参照ラベルによってセグメント・データの供給源が符号化されている場合、そのラベルを、参照リゾルバが、参照付きセグメントを見つけるのを助けるのに用いることができる。

【００９２】

デコーダは、（段階S3または段階S7の後で）参照付きセグメントのセグメント・データにアクセスした後、セグメント・データを出力する（S8）。デコーダは次いで、他のトークンがあるどうか検査する（S9）。もっとトークンがある場合、段階2でプロセスが次のトークンによって繰り返され、そうでない場合、プロセスは完了する。

【００９３】

上記の説明は、PSSを用いてセグメント・バインディングおよび参照を符号化し復号化する1つの特定の態様のみを表している。McCanne IIに示されているようなラベルとデータとの関係のより精密な表現を伴う他の手法が可能である。

【００９４】

図8は、トランザクション加速が実現され、プロアクティブ・セグメント・ディストリビュータ（「PSD」）を使用する、ネットワーク化されたシステムのブロック図である。図8に示されているように、PSD 210は、PSDコントローラ212と、それ自体のPSS 214と、PSD変数用の他の記憶装置216とを含んでいる。いくつかの実現態様では、複数のPSDが用いられる。ただし、1つのPSDしか示されていない。

【００９５】

PSD 210の動作によって、各セグメントは必要なときに存在する可能性が高くなり、したがって、必要なセグメント変換要求が少なくなる。各セグメントをPSSからPSSに移動させる必要がある場合、PSD 210は、セグメントが実際に必要になる前にこのプロセスをトリガすることができ、したがって、トランザクションはより高速に戻る。これは、受信側TAが、ペイロードを受信する際にセグメントを求める要求を送信側TAに発行できるように遮断する必要がなくなるからである。PSD 210は、配信自体を行うか、または単にセグメントの所有者（または保持者）に配信するよう指示することができる。場合によっては、PSD 210は、それ自体のPSS 214を維持することができるが、いくつかの実現態様では、PSD 210は、単にバインディングの流れをPSS間に向け、それ自体のPSSは維持しない。

【００９６】

PSD 210は、CTA 20およびSTA 22からのトランザクションの流れを監視し、そのことから、どのセグメントが必要になる可能性が高いかおよびセグメントがどこで必要になるかを判定する。PSD 210は、あるセグメントが必要になると判定した場合、ファイル・システムまたはeメール・システムを実現するSTAなどの送信側TAにメッセージを送信することができる。このメッセージは、送信側TAにセグメント化を実行し、バインディングをそれ自体のPSSに記憶し、場合によってはバインディングを他のPSSに伝搬させ、したがって、セグメント化は、送信側TAにペイロードを送信させるメッセージを送信側TAが受信したときに行われる。セグメント化がうまく行われると、受信TAは、参照を含むペイロードを受信したときに必要なバインディングを得て、このようなバインディングは、帯域幅がそれほど重要でなくなったときに送信することができる。通常、送信側TAはSTAであるが、PSDはCTAにバインディングをシステムに「プレロード」しておくよう指示することもできる。

【００９７】

場合によっては、プリローディングの候補を識別するためにサーバにサーバ・エージェントが付加される。たとえば、Microsoft Exchange(商標)サーバなどのメール・サーバは、ネットワークに結合され、STAおよび関連するサーバ・エージェントと一緒に動作することができる。サーバ・エージェントは、過去の状況またはオペレータの方針に基づいて、eメールおよび添付ファイルがいつ到着するかを検出し、関連するセグメント・データを特定のCTAにプレロードする。これは、静的構成または好ましくは測定によって、どのユーザがeメールをどの位置から読み取るかを追跡することによって行うことができる

。次いで、リモート・ユーザがeメールを読むとき、大部分のeメール・データはすでにユーザのリモート・サイトに存在しているが、トランザクションはそれにもかかわらず、プロトコルが正しくなるようにExchangeメール・サーバに戻る。

【００９８】

PSD 210は、セグメントの生成をプロアクティブにトリガするだけでなく、参照が受信されたときにTAがセグメント・データを準備するようにすでに存在するバインディングを様々なTA PSSに「事前に存在させておく」のを助けることもできる。一実現態様では、PSD 210は、新しいバインディングがPSD 210に通知され、次いでPSD 210が新しいバインディングを通知側TAからすべてのまたはいくつかの他のTAに伝搬させ、さらにこれらのTAがバインディングを伝搬させるUSENETニュース項目と同様に伝搬モデルを処理する。バインディングを事前に存在させておくことをPSDによってトリガすることの代わりにあるいはそれに加えて、送信側TAは、どのセグメントを受信側TAに送信する必要があるかを予想し、受信側TAが未知のセグメントを変換するうえで追加的な要求を発行しなくても済むようにセグメントを早めに（または「帯域外で」）送信しておくことができる。

【００９９】

無差別の伝搬が、各トランザクションごとに完全な生データを送信することと比べてネットワーク・オーバロードまたは帯域幅使用度の増大を招く可能性が高い場合、より精密な手法を使用することができる。より効率的な手法の例では、PSDは、ヒューリスティックスを用いてどのTAがどのセグメントを必要とするかを判定する。他の手法では、PSDがどのCTAが「代理される（agented）」サーバからのどのセグメントを必要とするかを判定するのを可能にする高レベルの情報をPSDに与えるサーバ・エージェントをサーバが含む。

【０１００】

サーバ・エージェントを含むPSDの他の態様はある種のファイル・システム・ミラーリングを含む。この場合、サーバ・エージェントはファイル・システム活動を監視し、ファイル・システムに新しいデータが書き込まれるたびに、エージェントはPSDに1つまたは複数のCTAとの適切なセグメント・バインディングを複製するよう指示する。ユーザまたはオペレータによって定義されるポリシーは、ファイル・システム全体のデータを複製するか、それとも構成済みの部分を複製するかを指示することができる。さらに、このようなポリシーは、ファイル・システムの最も頻繁にアクセスされる部分からのセグメント・データが複製されるように、アクセス・パターンの測定によって改良することができる（かつこのような測定はCTAごとに実行することができる）。その結果、そのような各CTAは、ファイル・システム・データのすべて（または一部）のミラーを含んでいる。次いで、クライアントがネットワーク・ファイル・システム・プロトコル（CIFSやNFSなど）を介してCTAと対話する際、トランザクションは元のファイル・サーバに戻るが、このようなトランザクションは全体的に、純粋な参照ストリングとして圧縮される。この手法では、すべてのクライアントが単一のファイル・サーバを共用する場合と同様に元のファイル・システム・セマンティクスが維持されるが、すべてのデータがクライアントごとのローカル・ファイル・サーバに存在する場合と同様にクライアント－サーバ通信が実行される。

【０１０１】

上記に概略的に説明したセグメント化はクライアント－サーバ通信の帯域幅要件を著しく低減させることができるが、トランザクションでは依然として、広い領域にわたって固有の通信呼出し時間が生じる。このような呼出し時間ボトルネックは、性能に悪影響を与える可能性があり、ファイル・リードアヘッドおよびライト・ビハインドのような補足技術を用いて対処することができる。しかし、データの圧縮およびステージングのために、リード・アヘッド技術およびライト・ビハンド技術は、すべてのデータがすでにCTAに存在しているため、ネットワーク上でオーバヘッドをほとんど生じさせないので極めて有効である。

【０１０２】

これらの手法はすべて、様々な種類のCTA/STA通信に帯域幅ポリシーを用いる方式で補

足することができる。たとえば、PSDにある種の帯域幅制限をかけてステージング・アルゴリズムの侵襲性を制限することができる。他の例では、帯域幅優先順位を様々な種類のステージング・データに適用することができる（たとえば、ファイル・システム・セグメント複製をeメール添付ファイル・セグメント複製よりも優先することができる）。

【０１０３】

図9は、本発明の態様によるネットワーク化されたピア・ツー・ピア・システムのブロック図である。図9に示されているように、様々なピアがピア・トランザクション・アクセルレータ（PTA）182を介して互いに対話する。ピア180同士は直接対話することができる。ただし、このような接続は図示されていない。動作時には、あるピア180が、各ピアのPTA 182およびネットワーク184を介してデータを要求することができる。図示のように、各PTA 182は、ピア・プロキシ190、TT 192、TT^{-1} 194、PSS 196、およびRR 198を含んでよい。ピア・ツー・ピア・システムでは、ピアは本質的に、あるトランザクションではクライアントとし機能し他のトランザクションではサーバとして機能し、したがって、トランザクション加速方式も同様に機能する。

【０１０４】

図10は、トランザクション加速が実現され、クライアント側トランザクション・アクセルレータが個別のエンティティとして存在するのではなくクライアントと一体化されるネットワーク化システムのブロック図である。図示のように、クライアント・システム302は、ネットワーク304を通してサーバ306に直接結合され、サーバ・トランザクション・アクセルレータSTA 310を介してサーバ308に結合されている。クライアント・システム302は、通信プロセス320と、直接ネットワーク入出力プロセス322と、CTAプロセス324と、持続セグメント・ストア328を含む記憶装置326とを含むように示されている。通信プロセス320は、直接ネットワーク入出力プロセス322、CTAプロセス324、および記憶装置326に結合されている。CTAプロセス324はPSS 328に結合されている。

【０１０５】

動作時には、通信システム320は、クライアント・システム302の外部のサーバとの対話を必要とする機能を、通常アプリケーション層で実行する。たとえば、通信プロセスは、webブラウザ、eメール・クライアント、Javaプログラム、および対話型ネットワーク・プログラム、チャット・プログラム、FTPプログラムなどを含んでよい。通信プロセスは、サーバと直接対話する場合、直接ネットワーク入出力プロセス322と対話してサーバと対話するが、トランザクションを加速する場合、通信プロセスはCTA 324と対話する。いくつかの態様では、通信プロセス320およびCTA 324は単一のアプリケーション・プログラムの構成要素であり、一方、他の態様では、別々のアプリケーション・プロセスであってよい。CTAプロセス324は、記憶装置326の一部をPSSとして用いて、上述のような様々な独立STAと同様にトランザクションを加速することができる。いくつかの変形態様では、PSS 328は、通信プロセス320に必要なプロセスのような、クライアント・システム302内の他のプロセスに用いられる、記憶装置326と異なるメモリである。

【０１０６】

直接ネットワーク入出力プロセス322は、ネットワーク304上のサーバと対話することによって通信プロセス302のネットワーク入出力要件を満たす。場合によっては、直接ネットワーク入出力プロセス322は、サーバ308への点線で示されている、CTA 324と同じサーバと対話する。クライアント・システム302は、トランザクション加速に関連するプロセスを含む、図示されていない他のプロセスを含んでよい。たとえば、通信プロセス320は、トランザクションをいつサーバに直接送信すべきかおよびいつトランザクションの加速を試みるべきかを判定する別個のプロセスに依存してよい。

【０１０７】

図11は、トランザクション加速が実現され、サーバ側トランザクション・アクセルレータがサーバと一体化されるネットワーク化システムのブロック図である。図11は、サーバ・システム352、ネットワーク354、クライアント356、クライアント358、およびクライアント・トランザクション・アクセルレータ（CTA）360を示している。サーバ・システム35

2は、通信プロセス370と、直接ネットワーク入出力プロセス372と、STAプロセス374と、持続セグメント・ストア378を含む記憶装置376とを含むように示されている。通信プロセス370は、直接ネットワーク入出力プロセス372、STAプロセス374、および記憶装置376に結合されている。STAプロセス374はPSS 378に結合されている。クライアント356は、直接クライアント356から、STAプロセス374を通過しないトランザクションに対処する直接ネットワーク入出力プロセス372までの線によって示されているようにサーバ・システム352に結合されている。クライアント358は、CTA 360およびSTAプロセス374を介してサーバ・システム352に結合されているが、他のトランザクションについては直接ネットワーク入出力プロセス374に直接接続することもができる。

【０１０８】

動作時には、通信システム370は、クライアントからの要求に応答するサーバ・プロセスなどの機能を実行する。サーバ・システム352とクライアントが直接対話する場合、トランザクションは直接ネットワーク入出力プロセス372を介して通信プロセスとクライアントとの間を流れる。STAプロセス374は、記憶装置376の一部をPSSとして用いて、上述のような様々な独立STAと同様にトランザクションを加速することができる。いくつかの変形態様では、PSS 378は、通信プロセス370に必要なプロセスのような、サーバ・システム352内の他のプロセスに用いられる、記憶装置376と異なるメモリである。

【０１０９】

直接ネットワーク入出力プロセス372は、ネットワーク354上のサーバと対話することによって通信プロセス352のネットワーク入出力要件を満たす。場合によっては、直接ネットワーク入出力プロセス372は、クライアント358への点線で示されている、STA 374と同じサーバと対話する。サーバ・システム352は、トランザクション加速に関連するプロセスを含む、図示されていない他のプロセスを含んでよい。たとえば、通信プロセス370は、トランザクションをいつサーバに直接送信すべきかおよびいつトランザクションの加速を試みるべきかを判定する別個のプロセスに依存してよい。

【０１１０】

内部CTAを有するクライアント・システムが内部STAを有するサーバ・システムと通信できるように図10および11の要素を組み合わせてよいことを理解されたい。また、単一の矢印線が使用されている場合、双方向情報流または双方向データ流が存在する可能性もあることを理解されたい。

【０１１１】

クライアントおよび／またはサーバ装置にTAを埋め込むことの1つの欠点は、各装置がそれ自体のPSSで終わり、所与の位置の多数のクライアント（またはサーバ）のために同じセグメント・データをキャッシングすることの利点が低減することである。しかし、この問題は、PSSが、好ましくは共通のLANセグメント（高速リンク、たとえば、建物または互いに近接した複数の建物内の複数の床を相互接続する高速キャンパス・エリア・ネットワークと相互接続される共通ネットワーク領域）上に位置する複数のTAに論理的に対応させる他の態様で解消することができる。この場合、論理共用PSSは、ネットワークに取り付けられた他の装置であっても、協働プロトコルによって（たとえば、IP Multicast上で）これらのPSSが単一の論理エンティティとして動作するように各CTAに埋め込まれた複数のPSSであってもよい。

【０１１２】

図12は、トランザクション加速が実現され、PSSが複数のトランザクション・アクセルレータ間で共用されるネットワーク化されたシステムのブロック図である。図12に示されているように、各クライアントはトランザクション加速ができるようにローカルCTA 402に結合されている。ローカルCTA 402は、別個のPSSを維持する代わりに、共用PSS 404に結合されている。好ましくは、ローカルCTAと共用PSSとの間の接続は、クライアントとサーバとの間に存在するネットワーク405を介した接続と比べて高い性能を有する接続である。共用参照リゾルバ406を存在させ、共用PSS 404およびそのPSSを共用するローカルCTAに結合してもよい。

【０１１３】
各ローカルCTA 402が、要求メッセージによってトランザクションを開始するか、または応答メッセージを受信すると、そのローカルCTA 402は、共用PSS 404を用いてセグメント・データを記憶し検索する。このことは、1つのローカルCTAのトランザクションの結果として記憶されるセグメントを別のローカルCTAのトランザクションに用いることができるという点で、各ローカルCTAに別個のPSSを用いるシステムに勝る利点を有する。たとえば、ローカルCTA 402(1)が、クライアントの、サーバSからデータを得ることを含むトランザクションに最近対処した場合、そのサーバがそのトランザクションのために作成したセグメントは、共用PSS 404に存在する可能性が高い。ローカルCTA 402(2)が、異なるクライアント（ラウンドロビン・ローカルCTA共用方式のようないくつかの構成では同じクライアント）の、サーバSを対象とするトランザクションに対処する場合、ローカルCTA 402(2)はサーバSのSTAに要求を送信する。第2のトランザクション用のセグメントが、ローカルCTA 402(1)との以前のトランザクションのセグメントと一致する場合、セグメントが実際には同じ要求を表すか、それとも結果として得られるペイロード・データがある共通のデータを有する無関係の要求を表すかにかかわらず、ローカルCTA 402(2)は、セグメント・データ自体ではなくこのようなセグメントの参照を受信する。

【０１１４】
ローカルCTAは、変換できないセグメントの参照を共用PSS 404から受信すると、変換を求める要求を共用参照リゾルバ406に送信することができる。いくつかの態様では、各ローカルCTAは、その変換された参照を共用PSS 404と、共用PSS 404が一構成要素であるローカルCTAの他の構成要素とに伝達するそれ自体の共用参照リゾルバを有している。他の態様は、すべてのクライアントによって使用される単一の共用参照リゾルバを使用することができる。

【０１１５】
共用PSSは図12では、クライアント側にあるように表されているが、同様の構成をサーバ側に設け、共用PSSまたは個々のPSSをクライアント側に設けてよい。さらに、共用PSSを含むTAが、個々のPSSを含むTAと同じネットワーク上に存在することができる。図12は共用PSS 404をローカルCTAと異なるものとして示しているが、共用PSSが1つのローカルCTA内に含まれてもよい。ただし、共用PSSは、そのPSSを共用する他のCTAの外部に位置する。

【０１１６】
PSSは、それがローカル化されたネットワーク・マルチキャスト通信を用いてサービスを提供するローカルCTA同士の間に接続することができる。この手法では、各トランザクション・アクセルレータは、ローカルにスコーピングされた公知のマルチキャスト・グループに加入する。ローカル化されたスコーピングを用いることによって、システムはローカル高速ネットワークによって接続されたトランザクション・アクセルレータのみがこの機構によって互いの調和をとるようにすることができる。各ホストは、このグループ（セッション・パケットを交換する他の構成済みグループ）に送信される定期的セッション・メッセージ・パケットを生成することができ、このグループに加入している他のトランザクション・アクセルレータまでの往復時間推定値の計算が可能になる。Floyd, S., et al., 「A Reliable Multicast Framework for Light-weight Sessions and Application Level Framing」 in IEEE/ACM Transactions on Networking、1997年12月、第5巻、第6号、pp. 784-803（以下「Floyd et al.」）に示されているような公知の技術をこの目的に使用することができる。このセッション・プロトコルは、グループ内のすべてのメンバーが互いの存在を知るのを可能にし、かつメンバーの和からグループのサイズを推定することもできる。

【０１１７】
このマルチキャスト構成を用いて、セグメント・データをキャッシングするシステムをいくつかの方法で改良することができる。ある手法では、トランザクション・アクセルレータは、新しいセグメント・バインディングを受信するたびに、ローカルにスコーピング

されたグループ内の他のすべてのトランザクション・アクセルレータにセグメント・バインディングをマルチキャストすることができる。これによって、上記に概略的に説明した、各クライアントが別個のPSSを有する問題が軽減される。なぜなら、ローカルな1組のトランザクション・アクセルレータ内の各PSSが互いのレプリカであり、任意の所与のデータ・セグメントがWAN接続上で1回しか送信されないからである。

【０１１８】

ネットワーク・マルチキャスト接続上の伝送の信頼性を確保するには、Floyd et al.およびFloyd et al.で引用された信頼できるマルチキャスト・プロトコルに関する論文のような、信頼できるマルチキャスト転送を行うためのいくつかの公知の方式を用いることができる。このマルチキャスト通信が均質の高速ローカル・エリア・ネットワークまたはキャンパス・エリア・ネットワークで行われると仮定すると、輻輳制御およびWANマルチキャストの難しい問題が完全に解消される。

【０１１９】

図13は、マルチキャスト通信を用いて共用PSSが更新され読み取られる、図12のシステムのマルチキャスト実現態様を示すブロック図である。図12に示されている構成と同様に、ローカルCTA 412は、各クライアントおよびネットワーク405に接続され、他のローカルCTAと共用PSS 414を共用する。共用RR 416は、共用PSS 414の各インスタンス（414(1), 414(2), ...として示されている）と同じマルチキャスト・グループ417上に存在するように構成される。論理的には、マルチキャスト・グループは、ローカルCTAが、共用PSSを読み取り書き込むのに必要な入出力に対処する場合、共用RR 416およびローカルCTAを含むということができる。マルチキャスト・トラフィックは、図に線418で示されている。

【０１２０】

他の手法では、PSSが、上述とは異なりプロアクティブには複製されず、トランザクション・アクセルレータが、未知のセグメントの変換を求めるローカル要求を発行することができる。すなわち、トランザクション・アクセルレータは、そのPSSにないデータの参照を受信すると、ローカルにスコーピングされたマルチキャスト・グループ上で変換要求メッセージを送信する。したがって、他のすべてのトランザクション・アクセルレータは、エラーが起こらないかぎり要求メッセージを受信する。要求されたデータをPSS内に有する受信側は、このデータで応答することができる。（Floyd et al.のような）公知のスロッティング技術およびダンピング技術を用いることによって、通常、ネットワーク上で送信される応答メッセージは1つだけであり、しかも遅延はほとんど起こらない。

【０１２１】

（セッション・メッセージ往復時間から算出されるある故意の遅延の後）要求側は、応答を受信しなかった場合、データがローカル環境には存在しないと仮定し、最初に関心対象のデータ参照を生成したトランザクション・アクセルレータにWAN上で変換要求を送信する。ローカル往復時間は、WAN往復時間（通常10ミリ秒以上）と比べてかなり短い（通常1ms未満）ので、この初期検査によって起こる余分な遅延は無視することができ（通常、数パーセント以下）、一方、ローカル・ネットワーク性能が高くなるために実質的な利点がもたらされる。

【０１２２】

他の手法では、上述の2つの手法を混成することによって、ローカル変換要求に関連する遅延が無くなる。この混成手法では、トランザクション・アクセルレータは、新しいセグメント・バインディングを受信するたびに、セグメント全体をマルチキャストするのではなく、単にセグメントの名前をマルチキャストする。このように、すべてのローカル・トランザクション・アクセルレータは、すべてのセグメント・データのコピーを必ずしも保持する必要なしにどんなセグメントが存在するかを知る。次いで、PSSにはないが名前がローカルに既知の名前として記録されているセグメントの参照を受信すると、トランザクション・アクセルレータはそのデータを求めるローカル要求を送信することができ、ローカル要求は、送信側を識別できる場合、新しいセグメント・バインディングを送出したトランザクション・アクセルレータに直接アクセスすることができる。送信側を識別でき

ない場合、アクセルレータは、データがローカルには存在しないと仮定し、直ちにWANを介して要求を送信することができる。セグメントがローカルに存在すると推定されるときでも、そのセグメントが他のすべてのローカル・アクセルレータのPSSからフラッシュされている可能性がある。この場合、要求側アクセルレータは、これにもかかわらずタイムアウトし、WANを介したその変換要求の送信に戻る。

【０１２３】

他の手法では、ローカル・アクセルレータ・グループのPSSを介して記憶されているセグメント・データを完全に複製しなくてもよい。この場合、各アクセルレータは、協働キャッシング技術を用いてセグメント・キャッシュの一部に責任を負う。上述のように、他のアクセルレータによって管理されているセグメント・データの参照を変換する必要があるときは、要求をその装置に直接送信することも、間接的にマルチキャスト・グループ上で送信することもできる。データが再アセンブルされクライアント（またはサーバ）に供給された後、このデータを破棄することができ、ローカルPSSに入力する必要がなくなる（このセグメント・データが他のトランザクション・アクセルレータによってアクティブに管理されているため）。

【０１２４】

図14は、一体化されたCTAを含む複数のクライアント502を示している。クライアント502は、クライアント502をLAN-WANリンク508を介してWAN 506に結合するLAN 504に結合されている。LAN 504上のすべてのクライアントがCTA 512を含む必要があるわけではなく、少なくとも2つのクライアントが、一体化されたCTA 512を含むように示されている。各CTAは、PSS 514およびRR 514を含むように示されている。この実現態様では、CTAのすべての機能を、クライアント上で実行されるソフトウェアとして実現することができる。

【０１２５】

クライアントのCTA 512は、このクライアント上で実行されるクライアント・アプリケーション510によって要求されたトランザクションの加速に対処する。たとえば、クライアント502(2)上で実行されるアプリケーションが、加速すべきサーバとのトランザクションを開始する際、CTA 512(2)との接続が確立される。CTA 512(2)は次いで、LAN 504およびWAN 506上で、上述のように対応するSTAとの接続をオープンする。CTA 512(2)は、加速されたペイロードを含む応答メッセージを受信すると、PSS 514(2)のコンテントを用いて、加速されたペイロードの参照ラベルを参照解除する。

【０１２６】

LAN 504上のサーバと他のクライアントとの間に得られたであろうセグメントの利点を実現するために、各PSS 514は協働PSSであってよい。各CTAは協働することによって、それ自体のPSSとLAN 504上の他のCTAのPSSから得たセグメント・バインディングを使用することができる。次いで、セグメント・バインディングをローカルに見つけることができない場合、CTAのRRは、バインディングを求める要求をWAN上でSTAに送信することができる。

【０１２７】

場合によっては、RRは、新しいバインディングを受信すると（またはそのCTAが新しいバインディングを作成すると）、その新しいバインディングをLAN上の他の各RRに配信し、したがって、LAN上で作成された利用可能なバインディングが各クライアントのPSSに存在し、CTAはすでに、CTAがペイロードを参照解除するときにLAN上で利用可能な各バインディングのコピーを有している。これを本明細書では「規定協働（prescriptive cooperation）」と呼ぶ。

【０１２８】

他の場合には、バインディングは早めに配信されず、要求に応じて送信される。したがって、RRは、バインディングが必要であるとき、LAN上の他のRRにそのバインディングを要求する。これを「オンデマンド協働（on-demand cooperation）」と呼ぶ。

【０１２９】

これらの手法を混成した手法では、RRが新しいバインディングを受信するかまたはその

CTAが新しいバインディングを作成すると、RRは、新しいセグメントの参照および送信側CTAを示す「バインディング通知」をLAN上の他のCTAに配信する。他のCTAが、必要なバインディングをそれ自体のPSSに有していないと判定すると、CTAのRRは、すでに受信されているバインディング通知のリストを検査する。必要なバインディングがリストにある場合、要求側RRは、送信元CTAにメッセージを出してバインディングを得る。RRは、バインディングがなく、かつLAN上の他のCTAからのバインディング通知もないと判定した場合、バインディングを求める要求をWAN上で送信する。これを「通知協働（notice corporation）」と呼ぶ。

【０１３０】

所与のLANが上述の協働方式のうちの複数を実現できることを理解されたい。協働するRR間のメッセージ送信は、マルチキャスティングを用いて行うことができる。たとえば、各協働クライアント（またはそのCTAもしくはRR）はマルチキャスト・グループのメンバーであってよい。規定協働では、各送信側CTAが、受信または作成した新しいバインディングをマルチキャストする。オンデマンド協働では、要求側RRが要求をマルチキャストすることができ、応答側CTAがその回答をユニキャストまたはマルチキャストすることができる。回答をマルチキャストすることによって、バインディングを要求しなかった他のCTAがそのバインディングを受信し、場合によってはこのCTAのPSSに記憶することが可能になる。通知協働では、通知をマルチキャストすることができるが、要求の場合、要求側が、どのCTAが要求されたバインディングを有するかを知っているので、要求をユニキャストすることができる。もちろん、バインディング通知が送信側CTAを示さないか、またはその情報が記憶されない通知協働を実現することができる。この場合、バインディング要求をマルチキャストすることができるが、通知協働を使用する際の好ましい手法は、どのCTAがその通知を送信するかを追跡することである。

【０１３１】

図15は、トランザクション加速が実現され、ネットワークが様々なプロトコルおよびサービスに対処する、ネットワーク化されたシステムのブロック図である。CTAおよびSTAは、CIFSトランザクション、NFSトランザクション、SMTPトランザクション、IMAPトランザクション、およびHTTPトランザクションを加速するように結合されるように示されている。他の構成では、サーバが可変位置に位置し、クライアントが可変位置に位置する。それぞれの場合に、加速されたプロトコルのトランザクションは、CTAおよびSTAを通過し、上述のように加速することができ、トランザクションに関わるクライアントおよびサーバに対して透過的であってよい。図示のオープン・プロトコルだけでなく、CTAおよびSTAは、Microsoft Exchange(商標)、Lotus Notes(商標)などの所有権付きプロトコルのトランザクションを加速することができる。本明細書で説明した他の変形態様と同様に、TAをクライアントおよびサーバと一体化することができる。たとえば、ソフトウェア・ベンダによっては、その一連のクライアント－サーバ・ソフトウェアの一部としてトランザクション加速を含んでよい。

【０１３２】

上記の説明は、例示的なものであって制限的なものではない。当業者には、本開示を検討することにより本発明の多数の変形態様が明らかになると思われる。したがって、本発明の範囲は、上記の説明を参照して判定すべきではなく、その代わりに、添付の特許請求の範囲をその全範囲の均等物と共に参照して判定すべきである。

【図面の簡単な説明】

【０１３３】

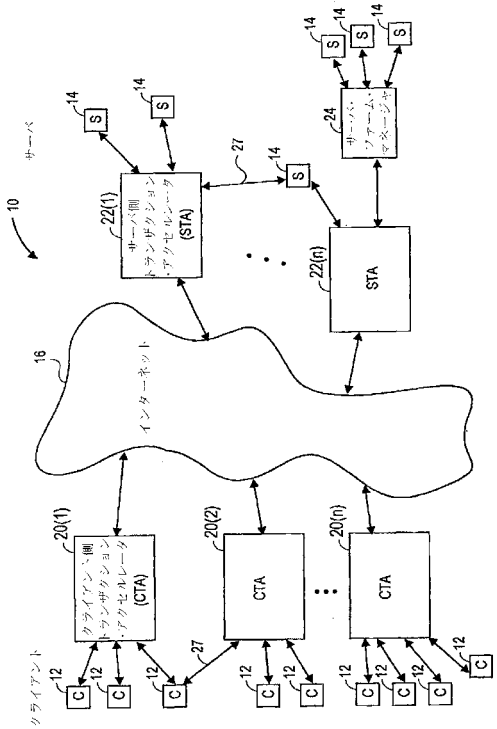
【図１】本発明の態様によるネットワーク化されたクライアント－サーバ・システムのブロック図である。

【図２】クライアント側トランザクション・アクセルレータ（「CTA」）およびサーバ側トランザクション・アクセルレータ（「STA」）を詳しく示し、図示の都合上、システム全体を簡単に示す、図1のシステムのブロック図である。

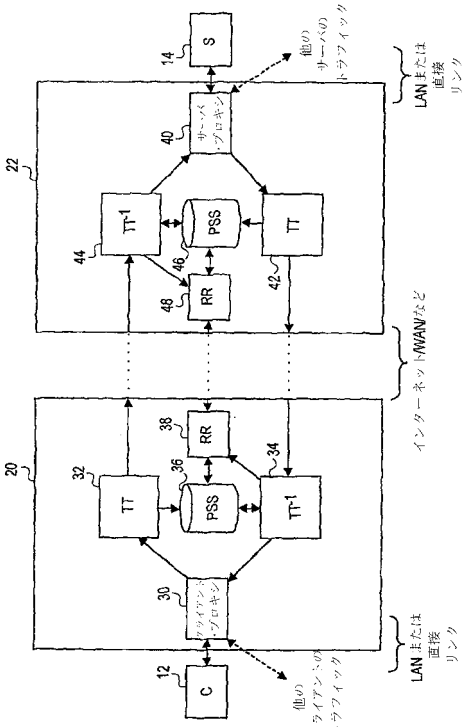
【図３】図1に示されているシステムと共に使用できる持続セグメント・ストア（「PSS」

）の態様におけるデータ構成の図である。
【図４】図2のトランザクション変換器（「TT」）で使用できるエンコーダのブロック図である。
【図５】図2の逆トランザクション変換器（「TT^{-1}」）で使用できるデコーダのブロック図である。
【図６】入力データがデータ・セグメントの参照によってセグメント化され表される符号化プロセスの図である。
【図７】図4のエンコーダによって出力できるデータを復号化するプロセスを示すフローチャートである。
【図８】トランザクション加速が、実施されプロアクティブ・セグメント・ディストリビュータ（「PSD」）を使用するネットワーク化システムのブロック図である。
【図９】本発明の態様によるネットワーク化されたピア・ツー・ピア・システムのブロック図である。
【図１０】トランザクション加速が実施され、クライアント側トランザクション・アクセルレータがサーバと一体化されるネットワーク化されたシステムのブロック図である。
【図１１】トランザクション加速が実施され、サーバ側トランザクション・アクセルレータがサーバと一体化されるネットワーク化されたシステムのブロック図である。
【図１２】トランザクション加速が実施され、PSSが複数のトランザクション・アクセルレータ間で共用されるネットワーク化されたシステムのブロック図である。
【図１３】マルチキャスト通信が共用PSSを更新し読み取るのに用いられる、図12のシステムのマルチキャスト実現態様を示すブロック図である。
【図１４】ローカルにLANを通してWANに結合された複数のクライアントのマルチキャスト実現態様を示すブロック図である。
【図１５】トランザクション加速が実施されネットワークが様々なプロトコルおよびサービスを取り扱うネットワーク化されたシステムのブロック図である。

【図１】



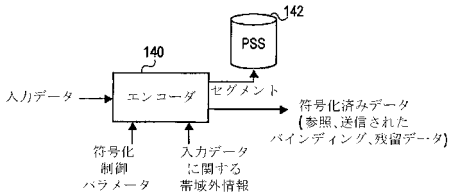
【図２】



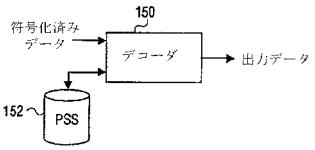
【図３】

参照データ		タイムスタンプIN	
R_1	セグメント1データ	00/00/0000, 00:00:00	•••
R_2	セグメント2データ	•••	•••
R_3	セグメント3データ	•••	•••
•••	•••		

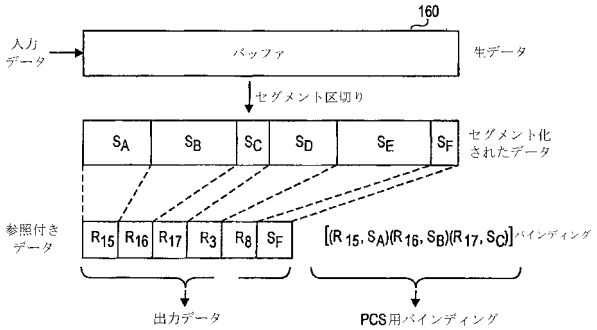
【図４】



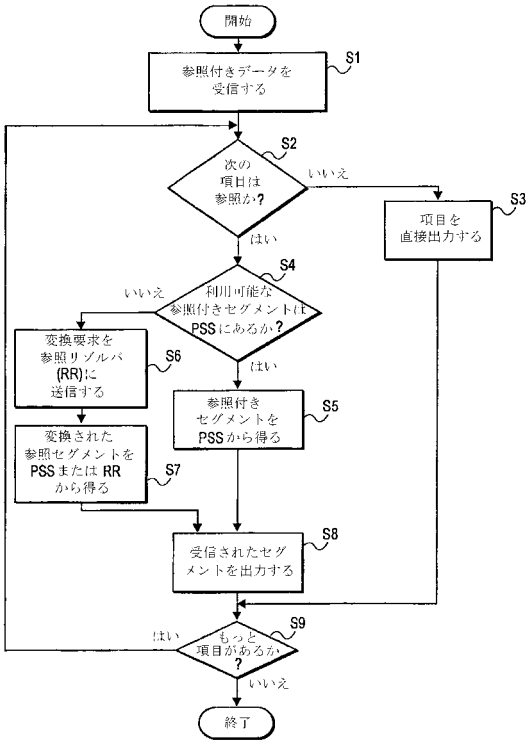
【図５】



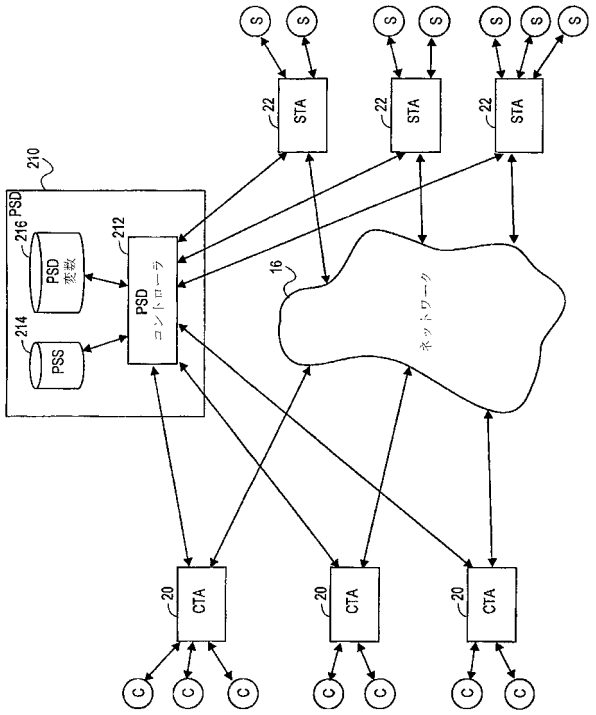
【図６】



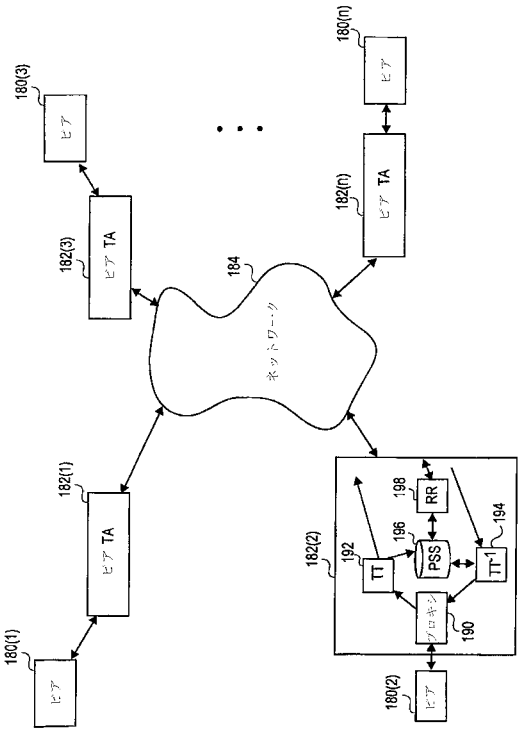
【図７】



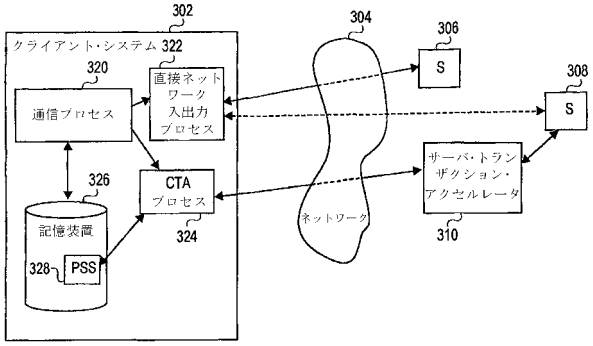
【図８】



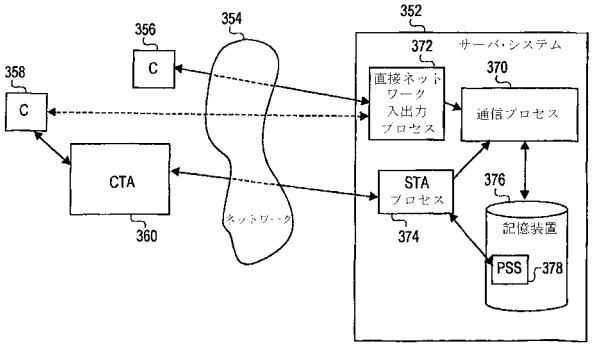
【図９】



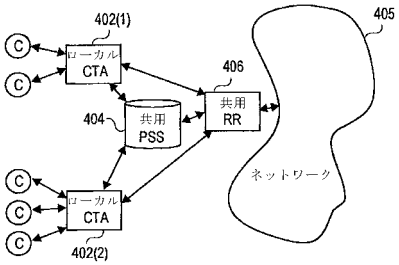
【図１０】



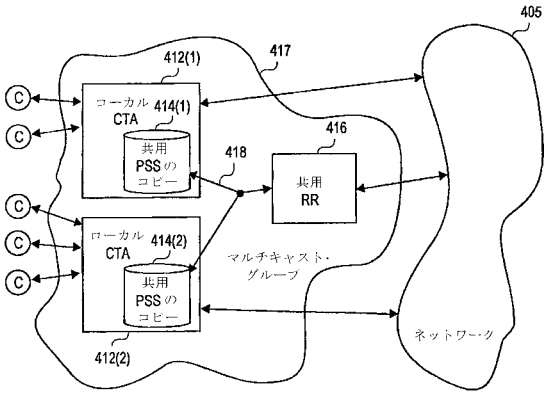
【図１１】



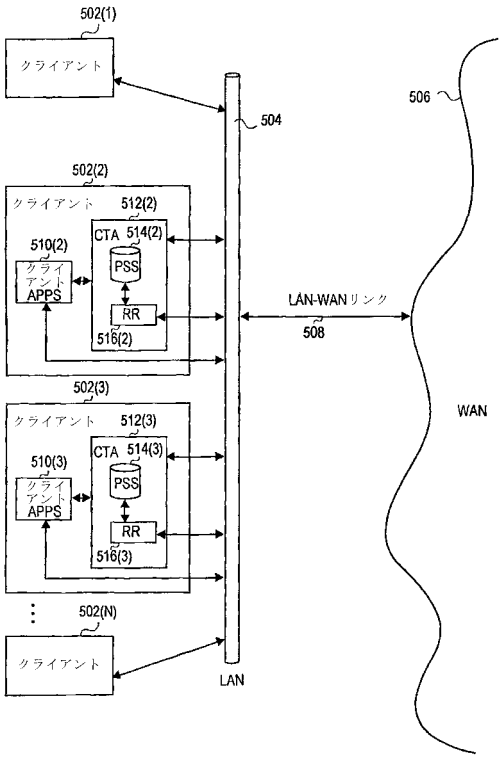
【図１２】



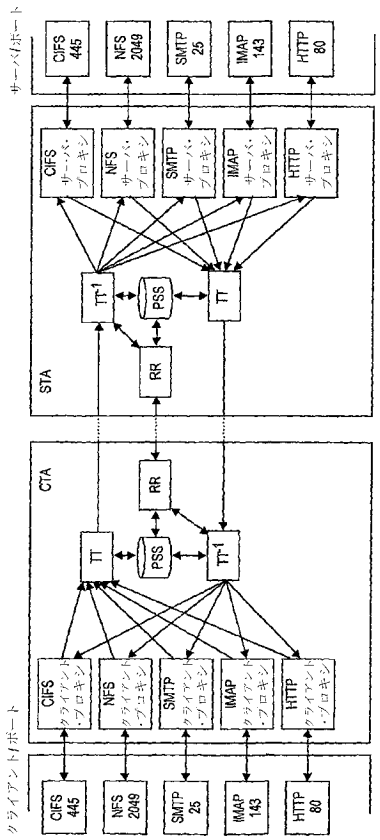
【図１３】



【図１４】



【図１５】


サーバ側ポート
CIFS 445
NFS 2049
SMTP 25
IMAP 143
HTTP 80
CIFS サーバ・プロキシ
NFS サーバ・プロキシ
SMTP サーバ・プロキシ
IMAP サーバ・プロキシ
HTTP サーバ・プロキシ
TT-1
PSS
TT
RR
STA
CTA
RR
TT
PSS
TT-1
CIFS クライアント・プロキシ
NFS クライアント・プロキシ
SMTP クライアント・プロキシ
IMAP クライアント・プロキシ
HTTP クライアント・プロキシ
クライアント側ポート
CIFS 445
NFS 2049
SMTP 25
IMAP 143
HTTP 80

【国際調査報告】

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US03/34232

A. CLASSIFICATION OF SUBJECT MATTER
IPC(7) : G06F 15/16
US CL : 709/203, 219
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
U.S. : 709/203, 219, 245, 246, 247, 213, 214, 215, 216; 341/55; 711/1, 147, 148

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
USPAT,USPAT;EPO;JPO;DERWENT;IBM-TDB USPAT

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
Please See Continuation Sheet

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 6,415,329 B1 (GELMAN) 02 July 2002 - see abstract col. 2 lines 58-63, col. 3 lines 8-19, col. 4, lines 10-21, col. 18, lines 13-22, col. 3 lines 1-19, col. 4, lines 1-21, 59-67.	1 - 10
Y	US 6,449,658 B1 (LAFE et al) 10 September 2002 - col. 6, lines 60-67, col. 6, lines 64-66, fig. 3, col. 2, lines 30-53, col. 3 lines 7-29, col. 5, lines 28-42.	1 - 10
Y	US 6,163,811 A (PORTER) 19 December 2000 - see abstract figs. 1A-1C & 4, col. 2 lines 1-8, col. 4 lines 6-23, col. 4 lines 51-65, col. 5, lines 16-31, col. 3 lines 29-37.	6 - 10
A,E	US 6,642,860 B2 (MEULENBROEKS) 04 November 2003 (08 January 2002).	1 - 10
A,P	US 6,553,141 B1 (HUFFMAN) 22 April 2003 (21 January 2000).	1 - 10

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
29 April 2004 (29.04.2004)	11 MAY 2004
Name and mailing address of the ISA/US Mail Stop PCT, Attn: ISA/US Commissioner for Patents P.O. Box 1450 Alexandria, Virginia 22313-1450 Facsimile No. (703) 305-3230	Authorized officer Vincent Trans Peggy Harod Telephone No. 703-305-3900

Form PCT/ISA/210 (second sheet) (July 1998)

INTERNATIONAL SEARCH REPORT

PCT/US03/34232

Continuation of B. FIELDS SEARCHED Item 3:
payload,bandwidth,england,transaction,transaction accelerat$3
segment,network,compress$5,partition$4,decompos$3frame,binding,storage,request,encod$3,table,database,server client,on-the-fly,dividing splitting,link pointer.

フロントページの続き

(81)指定国 AP(GH,GM,KE,LS,MW,MZ,SD,SL,SZ,TZ,UG,ZM,ZW),EA(AM,AZ,BY,KG,KZ,MD,RU,TJ,TM),EP(AT,BE,BG,CH,CY,CZ,DE,DK,EE,ES,FI,FR,GB,GR,HU,IE,IT,LU,MC,NL,PT,RO,SE,SI,SK,TR),OA(BF,BJ,CF,CG,CI,CM,GA,GN,GQ,GW,ML,MR,NE,SN,TD,TG),AE,AG,AL,AM,AT,AU,AZ,BA,BB,BG,BR,BY,BZ,CA,CH,CN,CO,CR,CU,CZ,DE,DK,DM,DZ,EC,EE,EG,ES,FI,GB,GD,GE,GH,GM,HR,HU,ID,IL,IN,IS,JP,KE,KG,KP,KR,KZ,LC,LK,LR,LS,LT,LU,LV,MA,MD,MG,MK,MN,MW,MX,MZ,NI,NO,NZ,OM,PG,PH,PL,PT,RO,RU,SC,SD,SE,SG,SK,SL,SY,TJ,TM,TN,TR,TT,TZ,UA,UG,US,UZ,VC,VN,YU,ZA,ZM,ZW

（特許庁注：以下のものは登録商標）
イーサネット
ＪＡＶＡ

(72)発明者 デマー マイケル ジェイ.
アメリカ合衆国 カリフォルニア州 サンフランシスコ ＃１５ フランクリン ストリート １８５

Ｆターム(参考) 5K030 GA03 HC01 JA05 KA04

【要約の続き】
グメントを、その持続セグメント記憶装置に記憶されているセグメントと比較し、置き換えるべきデータのセグメントと一致するかまたはほぼ一致するその持続セグメント記憶装置内のエントリの参照でデータのセグメントを置き換える。受信側トランザクション・ストアは、セグメント参照をその持続セグメント記憶装置からの対応するセグメント・データで置き換え、必要に応じて欠けているセグメントを送信側に要求することによって、送信されたデータを再構成する。トランザクション・アクチュエータは、複数のクライアントおよび／または複数のサーバを取り扱うことができ、持続セグメント・ストアに記憶されているセグメントをそれぞれの異なるトランザクション、それぞれの異なるクライアント、および／またはそれぞれの異なるサーバに関係付けることができる。持続セグメント・ストアには、他のトランザクション・アクセルレータから得たセグメント・データを存在させることができる。