

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第5948111号  
(P5948111)

(45) 発行日 平成28年7月6日(2016.7.6)

(24) 登録日 平成28年6月10日(2016.6.10)

(51) Int. Cl. F I  
G06F 13/00 (2006.01) G06F 13/00 520C

請求項の数 13 (全 26 頁)

(21) 出願番号	特願2012-85993 (P2012-85993)	(73) 特許権者	000005108 株式会社日立製作所 東京都千代田区丸の内一丁目6番6号
(22) 出願日	平成24年4月5日(2012.4.5)	(74) 代理人	100100310 弁理士 井上 学
(65) 公開番号	特開2013-218387 (P2013-218387A)	(74) 代理人	100098660 弁理士 戸田 裕二
(43) 公開日	平成25年10月24日(2013.10.24)	(74) 代理人	100091720 弁理士 岩崎 重美
審査請求日	平成26年11月7日(2014.11.7)	(72) 発明者	大石 裕司 東京都国分寺市東恋ヶ窪一丁目280番地 株式会社日立製作所中央研究所内
		(72) 発明者	奥野 通貴 東京都国分寺市東恋ヶ窪一丁目280番地 株式会社日立製作所中央研究所内 最終頁に続く

(54) 【発明の名称】 パケット通信装置および方法

(57) 【特許請求の範囲】

【請求項1】

クライアントとサーバの間で前記クライアントからのリクエストに対して前記サーバがレスポンスを送信する形式の通信方法において、

前記クライアントと前記サーバとに接続され、前記クライアントおよび前記サーバの間の通信を中継するプロキシシステムが、

前記サーバから取得したファイル一覧を含むレスポンスを、前記クライアントに送信し、

前記クライアントからの、前記ファイル一覧に含まれる第一のファイルを指定する第一のリクエストの受信を契機として、

前記クライアントから、前記ファイル一覧に含まれる前記第一のファイル以外の第二のファイルを指定する第二のリクエストを受信しなくても、前記ファイル一覧に含まれる、前記第二のファイルをプリフェッチするための第三のリクエストを前記サーバへ送信し、

前記サーバから、前記第三のリクエストに対するレスポンスである前記第二のファイルをプリフェッチする

ことを特徴とする通信方法。

【請求項2】

請求項1の通信方法であって、

前記プロキシシステムは、サーバ側プロキシとクライアント側プロキシとを含み、

前記プリフェッチは、前記サーバ側プロキシで行い、

10

20

前記サーバ側プロキシは、  
一つ以上のファイル取得を要求する第三のリクエストを前記サーバへ送信し、  
前記第三のリクエストに対するレスポンスとしてプリフェッチした一つ以上の前記第二  
のファイルを前記クライアント側プロキシに送信し、  
 前記クライアント側プロキシは、  
 前記サーバ側プロキシから前記一つ以上の第二のファイルを受信して保存し、  
受信した前記第二のファイルの一つを取得するための前記第二のリクエストを前記クラ  
イアントから受信した場合に、保存している該当する前記第二のファイルの一つを前記ク  
ライアントに送信する  
 ことを特徴とする通信方法。

10

## 【請求項3】

請求項2の通信方法であって、  
 前記クライアント側プロキシは、  
 前記クライアントから前記第一のリクエストを受信した後に、前記クライアントから、  
前記第二のリクエストを受信した場合、  
前記第二のリクエストに対するレスポンスとなるべき前記第二のファイルを前記サーバ  
 側プロキシから受信していない場合に、前記第二リクエストを前記サーバ側プロキシに転  
送する  
 ことを特徴とする通信方法。

20

## 【請求項4】

請求項3の通信方法であって、  
 前記サーバ側プロキシは、  
 前記クライアント側プロキシから前記第二のファイルを指定する前記第二のリクエスト  
を受信した場合に、  
当該サーバ側プロキシが、前記第二のファイルを指定する前記第三のリクエストを未送  
信の場合に、前記第二のリクエストを前記第三のリクエストとして他のリクエストより優  
先的に前記サーバに送信する  
 ことを特徴とする通信方法。

## 【請求項5】

請求項1の通信方法であって、  
 前記プロキシシステムは、サーバ側プロキシとクライアント側プロキシとを含み、  
 前記クライアント側プロキシは、  
一つ以上のファイル取得を要求する第三のリクエストを前記サーバ側プロキシへ送信し  
、

30

前記サーバ側プロキシは、  
 前記第三のリクエストを前記サーバに転送し、  
 前記サーバから前記第三のリクエストに対するレスポンスとして一つ以上の前記第二の  
ファイルを受信し、  
前記一つ以上の第二のファイルを前記クライアント側プロキシへ転送し、  
 前記クライアント側プロキシは、  
 前記サーバ側プロキシから前記一つ以上の第二のファイルを受信して保存し、  
受信した前記第二のファイルの一つを取得するための前記第二のリクエストを前記クラ  
イアントから受信した場合に、保存している該当する前記第二のファイルの一つを前記ク  
ライアントに送信する  
 ことを特徴とする通信方法。

40

## 【請求項6】

請求項5の通信方法であって、  
 前記クライアント側プロキシは、前記第三のリクエストにより複数のファイル取得を要  
求する場合、  
前記第三のリクエストにより複数のファイル取得を要求する場合、

50

前記第三のリクエストとして、一つのファイル取得を要求するリクエストを送信し、当該リクエストに対するレスポンスである一つの前記第二のファイルを受信する前に、次のファイル取得を要求する次のリクエストを、前記第三のリクエストとして送信することを特徴とする通信方法。

【請求項 7】

請求項 2 の通信方法であって、  
前記クライアント側プロキシは、複数の前記クライアントに接続され、前記ファイル一覧に含まれ、プリフェッチされた前記第二のファイルを保存し、第一の前記クライアントがリクエストしたファイルと同一のファイルに対するリクエストを、第二の前記クライアントから受信した場合に、前記第二のクライアントに対して、保存した前記ファイルを送信することを特徴とする通信方法。 10

【請求項 8】

請求項 2 の通信方法であって、  
前記クライアント側プロキシは、複数の前記クライアントに接続され、前記サーバ側プロキシは、前記ファイル一覧に含まれプリフェッチされた前記第二のファイルを保存し、第一の前記クライアントがリクエストしたファイルと同一のファイルに対する、第二の前記クライアントからのリクエストを受信した場合に、前記第二のクライアントに対するレスポンスとして、保存した前記ファイルを送信することを特徴とする通信方法。 20

【請求項 9】

請求項 2 記載の通信方法であって、  
前記クライアント側プロキシ及び前記サーバ側プロキシは、WebDAV に従う通信であって、前記第一のリクエストは、PROPFIND であり、前記ファイル一覧は、前記ファイルの実体の場所を示す URL であって、「207 MultiStatus」というレスポンスに含まれ、前記第二のリクエストは、当該ファイルの実体の取得を要求するための GET リクエストである  
ことを特徴とする通信方法。 30

【請求項 10】

請求項 2 記載の通信方法であって、  
前記クライアント側プロキシ及び前記サーバ側プロキシは、HTTP プロトコルに従う通信であって、前記第一のリクエストが HTML ファイルに対する GET リクエストであり、前記前記ファイル一覧は、「200 OK」というレスポンスに含まれ、前記第二のリクエストは、当該ファイルの実体の取得を要求するための GET リクエストである  
ことを特徴とする通信方法。 40

【請求項 11】

クライアントとサーバ間で、前記サーバに格納される情報の通信制御を行うプロキシシステムであって、前記クライアントと通信するクライアント通信部と、前記サーバと通信するサーバ通信部と、通信制御部と、を有し、前記通信制御部は、前記クライアントからの、第一のリクエストに対して前記サーバから取得したレスポンスが示す、前記サーバに格納されている複数の情報のうち、一つの第一の前記情報の取得 50

要求を前記クライアントから受信したことを契機として、前記レスポンスが示す、前記第一の情報を含む複数の前記情報の取得要求を、前記サーバに送信し、

前記複数の情報取得要求に対するレスポンスである前記複数の情報を前記サーバから受信した場合、前記クライアントに当該複数の情報を送信することを特徴とするプロキシシステム。

【請求項 1 2】

請求項 1 1 記載のプロキシシステムであって、

前記通信制御部は、

バッファを備え、

WebDAVプロトコルに従う、前記クライアントからの送信される前記第一のリクエストであるPROPFINDに対する前記サーバからのレスポンスに基づいて、特定されるフォルダに含まれるファイルのURLを用いて、前記フォルダに含まれるいずれかのファイルの取得要求を前記クライアントから受信した場合、前記フォルダに含まれる他のファイルも前記サーバからプリフェッチして前記バッファに保存し、

前記クライアントからの前記他のファイルの取得要求に応じて、保存したファイルをクライアントに送信する

ことを特徴とするプロキシシステム。

【請求項 1 3】

請求項 1 2 記載のプロキシシステムであって、

前記クライアント通信部は、複数の前記クライアントと通信し、

前記通信制御部は、前記複数のクライアントから、PROPFINDを受信した場合、前記バッファに保存したファイルを当該複数のクライアントにそれぞれ送信し、当該ファイルを前記バッファから削除する

ことを特徴とするプロキシシステム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、パケット通信装置および方法に関する。

【背景技術】

【0002】

ウェブシステムにおいて、クライアント側で、サーバからの情報を表示しようとする場合、クライアントサーバ間で、ウェブブラウザはHTMLを解析し、オブジェクトを発見するたびにそのオブジェクトに対するGETリクエストをサーバに送信する。通常ブラウザから一つのサーバへの同時接続数は制限されているため、ウェブページの読み込みにはクライアント・サーバ間の往復が多数回行われることとなる。

【0003】

クライアントとサーバが大陸間海底ケーブルなどの遅延の大きなワイドエリアネットワーク (Wide Area Network: WAN) により接続されている場合、ハイパーテキスト転送プロトコル (Hyper Text Transfer Protocol: HTTP) による通信は、効率が著しく低下する。

遅延の大きなネットワークでは、クライアントがリクエストを送信してからサーバからレスポンスが返ってくるまで長い時間がかかり、クライアントはその間待機する必要を生じる。これが多数回行われるためウェブページをリクエストしてから表示が完了するまでの時間が長くなる。

【0004】

例えば、クライアントとサーバが、衛星回線を通じて通信し、HTTPプロトコルに従って、サーバから衛星回線を介してダウンロードされるウェブページのインラインオブジェクトの呼び出し遅れを防ぐために、分散プロキシサーバが、基本コンポーネントの取得時に、インラインオブジェクトを先読みする技術が特許文献 1 に開示されている。

【0005】

また、HTTP方式を拡張して、サーバ上のファイルを容易に編集可能としたWebDAV (

10

20

30

40

50

Web Distributed Authoring and Versioning) という方式がある (非特許文献 1)。WebDAVではサーバ上のファイルやフォルダをWebDAVクライアント経由で操作可能である。WebDAVでもHTTPと同様のリクエストが用いられるほか、フォルダ内のファイルに関するプロパティ情報を取得するためのPROPFINDというリクエストが用いられる。

【先行技術文献】

【特許文献】

【0006】

【特許文献 1】W01999008429A

【非特許文献】

【0007】

【非特許文献 1】IETF RFC2518 “HTTP Extensions for Distributed Authoring - WEBDAV”

10

【発明の概要】

【発明が解決しようとする課題】

【0008】

しかしながら、WebDAVではHTTPと異なり、ファイルが基本コンポーネントとそれに付随するインラインオブジェクトのような主従関係になっておらず、特許文献 1の技術を適用しても、ファイルの関係性を見出すことができないため先読みすることはできない。

【0009】

そこで、本発明では、WebDAVをWAN経由で利用する場合に、適切なファイル一覧の取得方法を実現することを、一目的とする。

20

【0010】

また、WebDAVではファイル一覧の情報を取得したとしても、次のリクエストが判明しないとそれらのファイルを転送すべきか否かを断定できないため、ファイルの転送が不要な場合にもファイルの転送を行ってしまう。そこで本発明では、必要な場合のみに複数のファイルを効率的に転送することを他の目的とする。また、WebDAVをWAN経由で利用する場合に適切なファイルの高速転送を実現することを他の目的とする。

【課題を解決するための手段】

【0011】

少なくともいずれか一の課題を解決するために、本発明の一態様は、クライアントと接続されるクライアント側プロキシと、サーバと接続されるサーバ側プロキシを含むシステムで、前記クライアントおよび前記サーバの間の通信を中継する通信方法であって、前記クライアントと前記サーバの間で前記クライアントからのリクエストに対して前記サーバがレスポンスを送信する形式の通信方法において、前記クライアントから送信される第一のリクエストで取得される複数ファイルのリストを含むレスポンスを、クライアントに送信し、前記クライアントから、前記リストに関連する第二のリクエストを受領した場合、前記クライアントからのリクエストを待たずに、前記サーバに対して前記リストに該当する複数ファイルを取得するためのリクエストを送信し、前記複数ファイル取得リクエストに対するレスポンスとして前記複数ファイルのリストに該当する前記複数ファイルの実体を、前記サーバからプリフェッチする、ことを特徴とする通信方法。

30

40

【0012】

また別の態様は、クライアント装置とサーバ装置間で、サーバに格納されるデータの通信制御を行うプロキシシステムであって、クライアントと通信するクライアント通信部と、サーバと通信するサーバ通信部と、前記クライアントからの、第一のリクエストに対するレスポンスに含まれる複数の情報のうち、少なくともいずれか一の情報の取得要求を前記クライアントから受信した場合、前記レスポンスに含まれる複数の情報の取得要求を、サーバに送信し、前記複数の情報取得要求に対するレスポンスを、前記サーバからの受信した場合、前記クライアントに当該レスポンスを送信する、送信制御部と、を有する態様である。

50

## 【発明の効果】

## 【0013】

HTTPやWebDAVに従う通信で、サーバからクライアントへの複数ファイルを効率的に転送する。

## 【図面の簡単な説明】

## 【0014】

【図1】本発明の実施例1における通信方式を表すシーケンス図である。

【図2】本実施例におけるネットワーク構成図である。

【図3】本実施例におけるクライアント側プロキシの内部構成を表す図である。

【図4】本実施例におけるサーバ側プロキシの内部構成を表す図である。

10

【図5】本実施例におけるクライアント側プロキシおよびサーバ側プロキシのデータ管理テーブルを表す図である。

【図6】本発明の実施例1におけるクライアント側プロキシの動作を表すフローチャートである。

【図7】本発明の実施例1におけるサーバ側プロキシの動作を表すフローチャートである。

【図8】本発明の実施例2における通信方式を表すシーケンス図である。

【図9】本発明の実施例2におけるクライアント側プロキシの動作を表すフローチャートである。

【図10】本発明の実施例2におけるサーバ側プロキシの動作を表すフローチャートである。

20

【図11】本発明の実施例3における通信方式を表すシーケンス図である。

【図12】本発明の実施例3におけるクライアント側プロキシの動作を表すフローチャートである。

【図13】本発明の実施例3におけるサーバ側プロキシの動作を表すフローチャートである。

【図14】本発明の実施例4における通信方式を表すシーケンス図である。

【図15】本発明の実施例4におけるクライアント側プロキシの内部構成を表す図である。

【図16】本発明の実施例4におけるサーバ側プロキシの内部構成を表す図である。

30

【図17】本発明の実施例4におけるクライアント側プロキシおよびサーバ側プロキシのデータ管理テーブルを表す図である。

【図18】本発明の実施例4におけるクライアント側プロキシの動作を表すフローチャートである。

【図19】本発明の実施例4におけるサーバ側プロキシの動作を表すフローチャートである。

## 【発明を実施するための形態】

## 【実施例1】

## 【0015】

図2は本実施例におけるネットワーク構成を表す図である。

40

## 【0016】

クライアント(11)と同一のLAN(13)内にクライアント側プロキシ(12)を設置し、クライアント側プロキシ(12)がWAN(14)との通信を行う。同様に、サーバ(16)と同一のLAN(17)内にサーバ側プロキシ(15)を設置し、サーバ側プロキシ(15)がWANとの通信を行う。クライアント側プロキシ(12)とサーバ側プロキシ(15)が協調して動作することにより、LANに比べて遅延の大きなWANでも性能劣化を抑えて通信を行うことを可能とする。

## 【0017】

本実施例では、クライアント側プロキシ(12)とサーバ側プロキシ(15)が協調して動作するシステムを説明するが、クライアント側プロキシ(12)あるいはサーバプロ

50

キシ(15)が他方のプロキシの動作を兼ねて、他方の処理を省略するプロキシシステムであってもよい。

【0018】

本発明の第一の実施例における通信シーケンスを図1に示す。図1では通信にかかわる、少なくとも一以上のクライアント(11)、クライアント側プロキシ(12)、サーバ側プロキシ(15)、およびサーバ(16)のシーケンスを示している。

【0019】

クライアントがサーバ上のフォルダにアクセスする際、まず、クライアントは、そのフォルダに対してPROPFINDというリクエスト(101)を発行する。このリクエストは、クライアント側プロキシ(12)で受信され、サーバ側プロキシ(15)へとWANを介して

10

【0020】

サーバ側プロキシ(15)は、PROPFINDを受信すると、サーバ(16)へさらに転送する(121)。PROPFINDを受信したサーバ(16)はそれに対するレスポンスを生成し、クライアントに向けてWAN側へ返送する(122)。このレスポンスにはフォルダ内のファイル一覧が含まれる。同レスポンスはサーバ側プロキシ(15)でクライアント側プロキシ(12)へ転送され(112)、サーバ側プロキシ(15)は、PROPFINDに対するレスポンスを解析し、フォルダ内のファイル一覧を取得する。さらにクライアント側プロキシ(12)でクライアント(11)へ転送される(102)。クライアント側プロキシ(12)でも同様にこのレスポンスを解析し、ファイル一覧を取得する。クライアント(11)はクライアント側プロキシ(12)およびサーバ側プロキシ(15)とは、別にこのレスポンスを解析してファイル一覧を取得する。

20

【0021】

ファイル一覧を取得したクライアント(11)は、各ファイルに対して順にリクエストを送信する。クライアント(11)から送信された最初のファイルに対するリクエスト(103)は、クライアント側プロキシ(12)によって転送され(113)、サーバ側プロキシ(15)によって転送されて(123)サーバ(16)によって受信される。サーバ(16)はリクエスト(123)に対するレスポンスとしてファイルを送信する(124)。

【0022】

まず、サーバ側プロキシ(15)の動作を説明する。レスポンス(124)を受信したサーバ側プロキシ(15)はこれをクライアント側プロキシ(12)へ転送する(114)。同時に、サーバ側プロキシ(15)はファイル一覧中の次のファイルに対するリクエストを、クライアント(11)に代わってサーバ(16)に送信する(125)。サーバ(16)はこれに対するレスポンスとしてファイルを送信する(126)。該当ファイル(ファイル一覧のうちいずれか一のファイル)を受信したサーバ側プロキシ(15)は、このファイルをクライアント側プロキシ(12)へ転送(115)するとともに、ファイル一覧に次のファイルがあればそれに対するリクエストを送信する(127)。サーバ側プロキシ(15)が同リクエストに対するレスポンスを受信(128)するとこれをクライアント側プロキシ(12)へ転送する(116)。サーバ側プロキシ(15)は、ファイル一覧に次のファイルがないので、ファイル一覧を破棄して動作を完了する。

30

40

【0023】

次に、クライアント側プロキシ(12)の動作を説明する。クライアント側プロキシ(12)はファイル1のレスポンスを受信(114)し、ファイル1を取得すると、ファイル1をクライアント(11)へ転送する(104)。クライアント(11)はファイル2に対するリクエストを送信する(105)。一方でファイル2のレスポンスはサーバ側プロキシ(15)より送信される(115)ので、クライアント側プロキシ(12)は、ファイル2に対するリクエストとレスポンスがそろった段階で、クライアント(11)に対してレスポンスを送信する(106)。同様の動作をファイル一覧中の全てのファイルに対して繰り返し(107, 116, 108)、ファイル一覧を破棄して動作を完了す

50

る。

【0024】

上記動作は、クライアント側プロキシ(12)、サーバ側プロキシ(15)の双方にてクライアント(11)およびサーバ(16)の組ごとに独立に行われる。クライアント(11)およびサーバ(16)の識別はパケットヘッダ中のIPアドレスを用いて行う。

【0025】

以上により、「PROPFIND」に対するレスポンスに含まれる「207 Multi-Status」に該当するファイルの指定を受けたの契機にサーバ側プロキシ15が、「207 Multi-Status」に対応する複数ファイルの読み出しを実行し、クライアント側プロキシ12に「200 OK」により、それぞれファイル1-3を送る例を示した。

10

【0026】

図3はクライアント側プロキシ(12)の内部構成を表す図である。サーバ側プロキシ(12)は、クライアント11とサーバ16の間の通信を仲介し、パケットを送受信する通信装置であって、クライアント11とWANの間にあるのが一例である。または、クライアント側の局所ネットワークのゲートウェイ装置であってもよい。クライアントは、例えば情報処理装置、無線情報処理端末である。局所ネットワークは、有線、無線少なくともいずれかで構成されてもよい。

【0027】

LAN側コネクション管理部(301)はクライアント(11)との通信を行う。クライアント(11)からのリクエストはリクエスト判定部(302)にて、リクエストを送信したクライアント(11)が識別される。また、そのリクエストがPROPFINDやGETなど、プリフェッチ動作の契機となるものかどうか判定される。

20

【0028】

プリフェッチ動作中でないとき、受信したリクエストはそのままリクエスト送信部(305)に送られ、サーバ側プロキシ(15)へ送信される。プリフェッチ動作中はリクエストは一旦データ管理部(304)に登録される。

【0029】

データ管理部(304)はIPアドレスで識別されるクライアント(11)ごとにテーブルを持ち、リクエスト・レスポンスの状態とそのデータを管理する。例えばクライアントが3台あれば3つのテーブル(311~313)を持つ。リクエストを受信すればそのパス名をキーとしてテーブルを検索して、リクエスト受信済みのフラグを立てる。またプリフェッチされたレスポンスを受信すればレスポンス受信済みのフラグを立てる。テーブルのエントリは定期的に調べられ、リクエスト受信済みかつレスポンス受信済みであれば、レスポンスをレスポンス送信部(303)を介して、クライアントへ送信する。リクエスト受信時にレスポンスが登録されていなければ、リクエストをリクエスト送信部(305)を介してサーバ側プロキシ(15)へ送信してもよい。

30

【0030】

リクエスト送信部(305)は、リクエスト判定部(302)またはデータ管理部(304)から受け取ったリクエストをWAN側コネクション管理部(307)を用いて送信する。

40

【0031】

WAN側コネクション管理部(307)は、WANを通じてサーバ側プロキシ(15)との通信の制御を行う。

【0032】

レスポンス解析部(306)ではサーバ側プロキシ(15)から受信したレスポンスのIPアドレスを解析して、どのクライアント(11)に対するものであるか判断する。また、PROPFINDに対するレスポンスであればその内容も解析しファイル一覧を取得してデータ管理部(304)に登録する。同時にレスポンス送信部(303)に渡してクライアント(11)へ転送する。GETに対するレスポンスであればデータ管理部(304)に登録す

50

る。

【0033】

レスポンス送信部(303)ではレスポンス解析部(306)またはデータ管理部(304)から受信したレスポンスデータをLAN側コネクション管理部(301)に渡してクライアント(11)へ転送する。

【0034】

図4はサーバ側プロキシ(15)の内部構成を表す図である。サーバ側プロキシ(15)は、クライアントとサーバの間の通信を仲介し、パケットを送受信する通信装置であって、サーバとWANの間にあるのが一例である。または、サーバ側の局所ネットワークのゲートウェイ装置であってもよい。サーバは、例えば情報処理装置である。サーバ側の局所ネットワークは、有線、無線少なくともいずれかで構成されてもよい。WAN側コネクション管理部(401)はクライアント側プロキシ(12)との通信を行う。

10

【0035】

クライアント側プロキシからのリクエストはリクエスト判定部(402)にて、リクエストを送信したクライアント(11)が識別される。また、そのリクエストがPROPFINDやGETなど、プリフェッチ動作の契機となるものかどうか判定される。

【0036】

プリフェッチ動作中でないとき、受信したリクエストはそのままリクエスト送信部(405)に送られ、サーバへ送信される。プリフェッチ動作中の場合、リクエストは、一旦データ管理部(404)に登録される。

20

【0037】

データ管理部(404)は、クライアント側プロキシ(12)と同様にクライアント(11)ごとにデータ管理テーブルを持ち、リクエスト・レスポンスの状態とそのデータを管理する。例えばクライアント(11)が3台あれば3つのテーブル(411~413)を持つ。プリフェッチ動作中は、テーブルは定期的に調べられ、プリフェッチ未了のエントリがあればプリフェッチを行うため、リクエスト送信部(405)へURLを渡す。クライアント側プロキシ(12)からリクエストを受信した場合、該当レスポンスを送信していなければ、そのリクエストを優先的にリクエスト送信部(405)へ渡し、サーバ(16)へ送信する。

【0038】

リクエスト送信部(405)は、リクエスト判定部(402)またはデータ管理部(404)から受け取ったリクエストをLAN側コネクション管理部(407)を用いて送信する。LAN側コネクション管理部(406)は、サーバ側プロキシ(15)とサーバ(16)との通信を制御する。

30

【0039】

レスポンス解析部(406)ではサーバ(15)から受信したPROPFINDに対するレスポンスを解析しファイル一覧を取得してデータ管理部(404)に登録する。同時にレスポンス送信部(403)を介してクライアント(16)へ転送する。またレスポンスがGETに対するレスポンスであればデータ管理部(404)に登録する。

【0040】

レスポンス送信部(403)は、レスポンス解析部(406)またはデータ管理部(404)から受信したレスポンスデータをWAN側コネクション管理部(401)を介してクライアント側プロキシ(12)へ転送する。

40

【0041】

図5は、(a)クライアント側プロキシおよび(b)サーバ側プロキシのデータ管理部における、データ管理テーブル(311, 411)を示す。クライアント側プロキシのデータ管理テーブル(311)は対象となるファイルごとに、ファイルのURL(351)、プリフェッチしたレスポンス(352)、および1または0の値をとる3種類のフラグを持っている。フラグは「クライアントからリクエスト受信済み(353)」、「サーバ側プロキシからレスポンス受信済み(354)」、「クライアントにレスポンス送信済み(355)」

50

」である。

【 0 0 4 2 】

「クライアントからリクエスト受信済み ( 3 5 3 ) 」フラグは、そのファイルに対するリクエストをクライアントから受信したとき1に設定される。フラグの値は、0 , 1 以外であっても、クライアントからのリクエストが受信されたか否かを、識別できる値を用いてもよい。

【 0 0 4 3 】

「サーバ側プロキシからレスポンス受信済み ( 3 5 4 ) 」フラグはそのファイルをサーバ側プロキシから受信したとき1に設定される。フラグの値は、1 以外であっても、サーバ側プロキシからレスポンスが受信されたか否かを、識別できる値を用いてもよい。これらのフラグの両方が1となっていれば、レスポンスはクライアント側プロキシからクライアントへ送信され、「クライアントにレスポンス送信済み ( 3 5 5 ) 」フラグが1に設定される。「クライアントにレスポンス送信済み ( 3 5 5 ) 」フラグが1であれば該当エントリは削除される。

10

【 0 0 4 4 】

サーバ側プロキシのデータ管理テーブルは対象となるファイルごとに、ファイルのURL ( 4 5 1 )、プリフェッチしたレスポンス ( 4 5 2 )、および1または0の値をとる3種類のフラグを持っている。フラグは「サーバにプリフェッチリクエスト送信済み ( 4 5 3 ) 」、「クライアント側プロキシからリクエスト受信済み ( 4 5 4 ) 」、「サーバからレスポンス受信済み ( 4 5 6 ) 」である。

20

【 0 0 4 5 】

「サーバにプリフェッチリクエスト送信済み ( 4 5 3 ) 」フラグは、そのファイルに対するリクエストをサーバに送信したとき1に設定される。フラグの値は、0 , 1 以外であっても、サーバにプリフェッチリクエストが送信されたか否かを、識別できる値を用いてもよい。「クライアント側プロキシからリクエスト受信済み ( 4 5 4 ) 」フラグはクライアント側プロキシからリクエストを受信したときに1に設定される。「サーバからレスポンス受信済み ( 4 5 5 ) 」フラグはサーバからレスポンスを受信したときに1に設定される。このフラグが1のときにレスポンスはクライアント側プロキシに送信され、エントリが削除される。

【 0 0 4 6 】

図6はクライアント側プロキシ ( 1 2 ) の動作を表すフローチャートである。初期状態ではLAN側でパケットを受信するまで待機する ( 5 0 1 )。クライアント側プロキシ ( 1 2 ) は、LAN側でパケットを受信すると、送信元および宛先IPアドレスごとに分離する ( 5 0 2 )。そのパケットがPROPFINDリクエストであれば ( 5 0 3 )、クライアント側プロキシ ( 1 2 ) は、サーバ側プロキシへ転送して ( 5 0 4 )、GET待ち状態へ移行する。その他のリクエストであれば ( 5 5 1 )、同様にサーバ側プロキシへ転送する ( 5 5 2 ) が、プリフェッチの準備は行わず、サーバ側プロキシ ( 1 5 ) からレスポンスを受信するのを待って ( 5 5 3 ) それをクライアント ( 1 1 ) へ転送し ( 5 5 4 )、初期状態 ( 5 0 1 ) へ戻る。

30

【 0 0 4 7 】

サーバ側プロキシへ転送したPROPFIND ( 5 0 4 ) のレスポンスが受信されるまで、クライアント側プロキシ ( 1 2 ) は、当該PROPFIND ( 5 0 4 ) に関する処理を待機する ( 5 0 5 )。PROPFIND ( 5 0 4 ) のレスポンスが受信されると、クライアント側プロキシ ( 1 2 ) は、そのレスポンスをクライアント ( 1 1 ) へ転送する ( 5 0 6 )。また、クライアント側プロキシ ( 1 2 ) は、このレスポンスを解析してファイル一覧を作成し、データ管理部 ( 4 0 4 ) に登録する ( 5 0 7 )。

40

【 0 0 4 8 】

続いて、クライアント側プロキシ ( 1 2 ) は、クライアントからのGETリクエストを待ち受ける ( 5 0 8 )。GETを受信すると、クライアント側プロキシ ( 1 2 ) は、これをサーバ側プロキシ ( 1 5 ) へ転送し ( 5 0 9 )、GETリクエストに対するレスポンスを受

50

信するまで待機する(510)。クライアント側プロキシ(12)は、受信されればクライアント(11)へ転送(511)し、サーバ側プロキシ(15)からのプリフェッチデータを待ちうける(512)。GET以外のリクエストを受信した場合(555)は、同様にサーバ側プロキシ(15)へ転送する(556)が、WAN側からレスポンスを受信するのを待って(557)それをクライアント(11)へ転送し(558)、初期状態(501)へ戻る。

【0049】

サーバ側プロキシ(15)からのプリフェッチデータを受信すると(512)、クライアント側プロキシ(12)は、これをデータ管理部(304)に保存して(513)、クライアントからのリクエストを待ちうける(514)。リクエストを受信すると、リクエストに該当するファイルをデータ管理部(304)に保存されているプリフェッチデータから読み出し、保存していたプリフェッチデータをレスポンスとしてクライアントに送信する(515)。

10

【0050】

ファイル一覧にファイルが残っていれば、512以降を繰り返す(516)。すべてのデータを転送し終わるか、タイムアウトによりプリフェッチしたデータを破棄し初期状態(501)に戻る(517)。

【0051】

上記動作は、クライアント側プロキシ(12)、サーバ側プロキシ(15)の双方にてクライアント(11)およびサーバ(16)の組ごとに独立に行われる。クライアント(11)およびサーバ(16)の識別はパケットヘッダ中のIPアドレスを用いて行う。

20

【0052】

図7は、サーバ側プロキシ(15)の動作を表すフローチャートである。初期状態ではWAN側でパケットを受信するまで待機する(601)。サーバ側プロキシ(15)は、受信すると送信元および宛先IPアドレスごとに分離する(602)。そのパケットがPROPFINDリクエストであれば(603)、サーバ側プロキシ(15)は、サーバ(16)へ転送して(604)GET待ち状態へ移行する。その他のリクエストであれば(651)、サーバ側プロキシ(15)は、サーバ(16)へ転送する(652)が、プリフェッチの準備は行わず、サーバ側プロキシ(15)は、サーバ(16)からレスポンスを受信するのを待って(653)それをクライアント(11)側へ転送し(654)、初期状態(601)へ戻る。

30

【0053】

一方、サーバ側プロキシ(15)は、ステップ603の後、サーバ(16)へ転送したPROPFIND(604)のレスポンスを受信されるまで待機し(605)受信されるとクライアント側プロキシ(15)へ転送する(606)。また、サーバ側プロキシ(15)は、PROPFIND(604)に対するレスポンスを解析してファイル一覧を作成し、データ管理部(404)に登録する(607)。

【0054】

続いてクライアント側プロキシ(12)からのGETリクエストを待ちうける(608)。GETを受信すると、サーバ側プロキシ(15)は、これをサーバ(16)へ転送し(609)、それに対するレスポンスを受信するまで待機する(610)。受信されればクライアント側プロキシ(12)へ転送し(611)、データ管理部(404)に登録されているファイル一覧に従って、次のファイルに対するプリフェッチリクエストを送信する(612)。GET以外のリクエストを受信した場合(655)は、サーバ側プロキシ(15)は、にサーバへ転送する(656)が、サーバ(16)からレスポンスを受信するのを待って(657)それをクライアント側プロキシ(12)へ転送し(658)、初期状態(601)へ戻る。

40

【0055】

一方、ステップ612に対してサーバ(16)からのプリフェッチデータを受信すると(613)、サーバ側プロキシ(15)は、クライアント側プロキシに送信する(614)

50

)。

【0056】

ファイル一覧に従って、サーバ側プロキシ(15)は、サーバから受信されていないファイルが残っていれば、612以降を繰り返す(615)。すべてのデータを転送し終わるか、タイムアウトによりプリフェッチしたデータを破棄し初期状態(601)に戻る(616)。

【0057】

上記動作は、クライアント側プロキシ(12)、サーバ側プロキシ(15)の双方にてクライアント(11)およびサーバ(16)の組ごとに独立に行われる。クライアント(11)およびサーバ(16)の識別はパケットヘッダ中のIPアドレスを用いて行う。

10

【実施例2】

【0058】

本発明の第二の実施例における通信シーケンスを図8に示す。図8では通信にかかわる、クライアント(11)、クライアント側プロキシ(12)、サーバ側プロキシ(15)、およびサーバ(16)のシーケンスを示している。本実施例の方式により、実施例1の場合と比べて同等の通信効率を保ちながら、WAN上の通信をWebDAVの標準プロトコルにのっとったものとする。それによりWAN上の通信がファイアウォール、Intrusion Detection System/Intrusion Protection System、Deep Packet Inspectionなどのセキュリティシステムにより、異常な通信であると判断されて遮断されることを防ぐことができる。

【0059】

20

手順701、702、711、712、721、722は、図1の手順101、102、111、112、121、122と同様にPROPFINDのリクエストとそのレスポンスと同様である。

【0060】

ファイル一覧を取得したクライアント(11)は、各ファイルに対して順にリクエストを送信する。クライアント(11)から送信された最初のファイルに対するリクエスト(703)は、クライアント側プロキシ(12)によって受信される。クライアント側プロキシ(12)は、このリクエストを解析して、PROPFINDリクエストに対するレスポンスであるファイル一覧中に含まれるファイル(ファイル1)であると判断すると、このリクエストをサーバ側プロキシ(15)へ転送する(713)。さらに、クライアント側プロキシ(12)は、クライアントからのリクエスト(707)で指定されるファイル(ファイル1)ではないが、ファイル一覧にある他のファイル(ファイル2、3)に対するリクエスト(714、715)も発行する。これらのリクエストは前のリクエストに対するレスポンスを待つことなくサーバ側プロキシ(15)へ送信される(714, 715)。

30

【0061】

サーバ側プロキシ(15)は、クライアント側プロキシ(12)からのリクエストを受信(713)すると、ファイル一覧中のファイルで、クライアントからの最初のリクエストで指定されたファイル(ファイル1)に対するリクエストを、クライアント(11)に代わってサーバ(16)に送信する(723)。サーバ(16)は、これに対するレスポンスとしてファイル1を送信する(724)。これを受信したサーバ側プロキシ(15)は、このファイルをクライアント側プロキシ(12)へ転送(716)する。

40

【0062】

また、サーバ側プロキシ(15)は、サーバ(16)からのファイル1の受信に応じて、受信したファイル1が含まれるファイル一覧に次のファイル(ファイル2)があればそれに対するリクエストを送信する(725)。サーバ側プロキシ(15)が同リクエストに対するレスポンスを受信(726)すると、これをクライアント側プロキシ(12)へ転送する(717)。さらに、次のファイルも、ステップ725、726、717と同様にプリフェッチを行うと(727, 728, 718)、サーバ側プロキシ(15)は、ファイル一覧に次のファイルがない場合、ファイル一覧を破棄して動作を完了する。このリ

50

クエストは前のリクエストに対するレスポンスを受信してから開始してもよいし、複数のリクエストを同時に送信してもよい。

【0063】

クライアント側プロキシ(12)は、ファイル1のレスポンスを受信(716)すると、これをクライアント(11)へ転送する(704)。クライアント(11)はファイル2に対するリクエストを送信する(705)。一方でファイル2のレスポンスは、サーバ側プロキシ(15)より受信される(717)ので、ファイル2に対するリクエストとレスポンスがそろった段階で、クライアント(11)に対してレスポンスを送信する(706)。同様の動作をファイル一覧中の全てのファイルに対して繰り返し(718, 707, 708)、ファイル一覧を破棄して動作を完了する。

10

【0064】

上記動作は、クライアント側プロキシ(12)、サーバ側プロキシ(15)の双方にてクライアント(11)およびサーバ(16)の組ごとに独立に行われる。クライアント(11)およびサーバ(16)の識別はパケットヘッダ中のIPアドレスを用いて行う。

【0065】

図9は第二の実施例におけるクライアント側プロキシ(12)の動作を表すフローチャートである。ステップ801ないし807、851ないし854は、図6のステップ501ないし507、551ないし554と同様である。

【0066】

続いてクライアントからのGETリクエストを待ち受ける(808)。クライアント側プロキシ(12)は、GETリクエストを受信するとこれを契機にファイル一覧中のファイルに対してプリフェッチリクエストを発行する(809)。このリクエストは前のリクエストに対するレスポンスを待たずに送信される。ファイル一覧中の全てのファイルに対してリクエストを送信するまでこれを繰り返す(810)。GET以外のリクエストを受信した場合(855)は、同様にサーバ側プロキシ(15)へ転送する(856)が、WAN側からレスポンスを受信するのを待って(857)それをクライアント(11)へ転送し(858)、初期状態(801)へ戻る。

20

【0067】

サーバ側プロキシ(15)から最初のリクエストに対するレスポンスを受信すると(811)、クライアント側プロキシ(12)は、これをクライアントへ転送する(812)。クライアント側プロキシ(12)は、サーバ側プロキシ(15)からのプリフェッチデータを受信すると(813)、これをデータ管理部(304)に保存して(814)、クライアントからのリクエストを待ち受ける(815)。リクエストを受信すると、クライアント側プロキシ(12)は、保存していたデータをレスポンスとしてクライアントに送信する(816)。

30

【0068】

ファイル一覧にファイルが残っていれば、812以降を繰り返す(817)。すべてのデータを転送し終わるか、タイムアウトによりプリフェッチしたデータを破棄し初期状態(801)に戻る(818)。上記動作は、クライアント側プロキシ(12)、サーバ側プロキシ(15)の双方にてクライアント(11)およびサーバ(16)の組ごとに独立に行われる。クライアント(11)およびサーバ(16)の識別はパケットヘッダ中のIPアドレスを用いて行う。

40

【0069】

図10は第二の実施例におけるサーバ側プロキシ(15)の動作を表すフローチャートである。ステップ901ないし907、951ないし954は、図7のステップ601ないし607、651ないし654と同様である。続いてクライアント側プロキシ(12)からのプリフェッチリクエストを待ち受ける(908)。プリフェッチリクエストは、ファイル一覧に含まれるファイルの数に対応して複数送られるため、サーバ側プロキシ(15)は、ファイル一覧に含まれるファイルの数に対応する全てのリクエストを受信するまでこれを繰り返す(909)。続いて、プリフェッチリクエストを順次サーバ(16)

50

へ転送し(910)、それに対するレスポンスを受信するまで待機する(911)。サーバ側プロキシ(15)は、レスポンスを受信されればクライアント側プロキシ(12)へ、それらを転送する(912)。GET以外のリクエストを受信した場合(955)は、サーバ側プロキシ(15)は、同様にサーバ(16)へ転送する(956)が、サーバ側プロキシ(15)は、サーバ(16)からGET以外のリクエストに対応するレスポンスを受信するのを待って(957)それをクライアント側プロキシ(12)へ転送し(958)、初期状態(901)へ戻る。

#### 【0070】

ファイル一覧にサーバから取得していないファイルが残っていれば、910以降を繰り返す(913)。すべてのデータを転送し終わるか、タイムアウトによりプリフェッチしたデータを破棄し初期状態(901)に戻る(914)。上記動作は、クライアント側プロキシ(12)、サーバ側プロキシ(15)の双方にてクライアント(11)およびサーバ(16)の組ごとに独立に行われる。クライアント(11)およびサーバ(16)の識別はパケットヘッダ中のIPアドレスを用いて行う。

10

#### 【0071】

装置内の構成については実施例1の場合と同様であるので、ここでは省略する。

#### 【0072】

以上の実施例2の説明により、WebDAVに従った通信により得られた情報を用いて、プリフェッチするリクエストとレスポンスとを対応づけ、プロキシ間での通信制御を行うことを説明した。その制御により、WAN上の通信がファイアウォール、Intrusion Detection System/Intrusion Protection System、Deep Packet Inspectionなどのセキュリティシステムにより、プリフェッチするための通信が、異常な通信であると判断されて遮断されることを防ぐことができる。

20

#### 【実施例3】

#### 【0073】

本発明の第三の実施例における通信シーケンスを図11に示す。この例では実施例2の方式をHTTPに適用している。本実施例の方式により、実施例2の場合と同様に、WAN上の通信はHTTPの標準ののりとしたものとなるため、セキュリティシステムにより異常な通信として判断され遮断されることを防ぐことができる。

#### 【0074】

クライアントが、サーバからウェブページを読み込む場合、まずHTMLファイルに対してGETというリクエスト(1001)を発行する。これはクライアント側プロキシ(12)で受信され、サーバ側プロキシ(15)へとWAN上を転送される(1011)。サーバ側プロキシ(15)はこれを受信すると、サーバ(16)へさらに転送する(1021)。HTMLに対するGETを受信したサーバ(16)は、そのHTMLファイルをクライアントへ返送する(1022)。このレスポンス(1022)は、HTTPに従った「200 OK」というレスポンスであり、フォルダ内のファイル一覧が含まれる。同レスポンスはサーバ側プロキシ(15)でクライアント側プロキシ(12)へ転送され(1012)、サーバ側プロキシ(15)は、このレスポンスを解析することによりフォルダ内のファイル一覧を取得する。さらにクライアント側プロキシ(12)でクライアント(11)へ転送される(1002)。クライアント側プロキシ(12)でも同様にこのレスポンスを解析し、ファイル一覧を取得する。クライアント(11)はクライアント側プロキシ(12)およびサーバ側プロキシ(15)とは、独立してこのレスポンスを解析してファイル一覧を取得する。

30

40

#### 【0075】

ファイル一覧を取得したクライアント(11)は、各ファイルに対して順にリクエストを送信する。クライアント(11)から送信された最初のファイルに対するリクエスト(1003)は、クライアント側プロキシ(12)によって受信される。

#### 【0076】

クライアント側プロキシ(12)は、このリクエストを解析して、ファイル一覧中に含

50

まれるファイルであると判断すると、このリクエストをサーバ側プロキシ(15)へ転送する(1013)。クライアント11からのリクエストで指定されるファイルが「200 OK」のレスポンス(1012)で取得したファイル一覧中に含まれるファイル(file1)であると、クライアント側プロキシ(12)が判断すると、そのファイル一覧の他のファイル(file2, 3)に対するリクエストも発行する。これらのリクエストは前のリクエストに対するレスポンスを待つことなくサーバ側プロキシ(15)へ送信される(1014, 1015)。つまり、クライアントからファイル一覧に含まれる一のファイルを指定するリクエストがくると、クライアント側プロキシ(12)は、ファイル一覧に含まれる他のファイルについてのリクエストも、クライアント(11)からのリクエストを待つことなく、サーバ側プロキシ(15)にプリフェッチリクエスト(1014、1015)を発行する。

10

**【0077】**

サーバ側プロキシ(15)は、クライアント側プロキシ(12)からのリクエストを受信(1013)すると、ファイル一覧中のファイルに対するリクエストを、クライアント(11)に代わってサーバ(16)に送信する(1023)。サーバ(16)はこれに対するレスポンスとしてファイルを送信する(1024)。

**【0078】**

これを受信したサーバ側プロキシ(15)は、このファイルをクライアント側プロキシ(12)へ転送(1016)するとともに、レスポンス(1022)で取得したファイル一覧に他のファイルがあればそれに対するリクエストをサーバ(16)に送信する(1025)。

20

**【0079】**

サーバ側プロキシ(15)は、同リクエストに対するレスポンスを受信(1026)すると、これをクライアント側プロキシ(12)へ転送する(1017)。他のファイルも同様にプリフェッチを行う(1027, 1028, 1018)。なお、サーバ側プロキシ(15)は、クライアント側プロキシ(12)から送信されるプリフェッチリクエストに対応するファイルのリクエストをサーバ(16)送信してもよい。サーバ側プロキシ(15)は、ファイル一覧に次のファイルがないので、ファイル一覧を破棄して動作を完了する。このリクエストは前のリクエストに対するレスポンスを受信してから開始される。

**【0080】**

30

クライアント側プロキシ(12)は、ファイル1のレスポンスを受信(1016)すると、これをクライアント(11)へ転送する(1004)。クライアント(11)は、ファイル一覧に含まれるファイル2に対するリクエストを送信する(1005)。一方でファイル2のレスポンスはサーバ側プロキシ(15)から送信される(1017)ので、クライアント側プロキシ(12)は、ファイル2に対するリクエストを保持し、レスポンスを受領した場合に、クライアント(11)に対して、保持しているファイル2に対するリクエストに対応するレスポンスを送信する(1006)。同様の動作をファイル一覧に含まれるファイルに対して繰り返し(1018, 1007, 1008)、ファイル一覧をバッファから破棄して動作を完了する。

**【0081】**

40

上記動作は、クライアント側プロキシ(12)、サーバ側プロキシ(15)の双方にてクライアント(11)およびサーバ(16)の組ごとに独立に行われる。クライアント(11)およびサーバ(16)の識別はパケットヘッダ中のIPアドレスを用いて行う。

**【0082】**

図12は、実施例3におけるクライアント側プロキシ(12)の動作を表すフローチャートである。初期状態ではLAN側でパケットを受信するまで待機する(1101)。受信すると送信元および宛先IPアドレスごとに分離する(1102)。そのパケットがHTMLに対するGETリクエストであれば(1103)サーバ側プロキシへ転送して(1104)GET待ち状態へ移行する。その他のリクエストであれば(1151)、同様にサーバ側プロキシ(15)へ転送する(1152)が、プリフェッチの準備は行わず、サーバ側プロキシ

50

(15) からレスポンスを受信するのを待って(1153)それをクライアント(11)へ転送し(1154)、初期状態(1101)へ戻る。

【0083】

サーバ側プロキシへ転送したHTMLに対するGET(1104)のレスポンスが受信されるまで待機し(1105)受信されるとクライアント(11)へ転送する(1106)。同時にこのレスポンスを解析してファイル一覧を作成し、データ管理部(304)に登録する(1107)。

【0084】

続いてクライアントからのGETリクエストを待ち受ける(1108)。ファイル一覧に含まれるファイルに対するGETリクエストを受信すると、クライアント側プロキシは、これを契機にファイル一覧中のファイルに対してプリフェッチリクエストを発行する(1109)。これらのリクエストは前のリクエストに対するレスポンスを待たずに送信される。また、クライアントからのGETリクエストで指定されるファイルだけでなく、GETリクエストで特定されるファイル一覧に含まれるファイルに対してリクエストを送信するまでこれを繰り返す(1110)。GET以外のリクエストを受信した場合(1155)は、同様にサーバ側プロキシ(15)へ転送する(1156)が、WAN側からレスポンスを受信するのを待って(1157)それをクライアント(11)へ転送し(1158)、初期状態(1101)へ戻る。

【0085】

サーバ側プロキシ(15)から最初のリクエストに対するレスポンスを受信すると(1111)、クライアント側プロキシ(12)は、これをクライアントへ転送する(1112)。サーバ側プロキシ(15)からのプリフェッチデータを受信すると(1113)、クライアント側プロキシ(12)は、これをデータ管理部(304)に保存して(1114)、クライアントからのリクエストを待ち受ける(1115)。クライアントからのリクエストを受信すると、クライアント側プロキシ(12)は、当該リクエストで指定されるファイルに対応する、ファイルを保存していたデータから読み出し、そのデータをレスポンスとしてクライアントに送信する(1116)。

【0086】

クライアント側プロキシ(12)は、ファイル一覧にファイルが残っていれば、1112以降を繰り返す(1117)。クライアント側プロキシ(12)は、すべてのデータを転送し終わるか、タイムアウトによりプリフェッチしたデータを破棄し初期状態(1101)に戻る(1118)。

【0087】

上記動作は、クライアント側プロキシ(12)、サーバ側プロキシ(15)の双方にてクライアント(11)およびサーバ(16)の組ごとに独立に行われる。クライアント(11)およびサーバ(16)の識別はパケットヘッダ中のIPアドレスを用いて行う。

【0088】

図13は、実施例3におけるサーバ側プロキシ(15)の動作を表すフローチャートである。初期状態ではWAN側でパケットを受信するまで待機する(1201)。受信すると送信元および宛先IPアドレスごとに分離する(1202)。そのパケットがHTMLに対するGETリクエストであれば(1203)サーバ(16)へ転送して(1204)GET待ち状態へ移行する。その他のリクエストであれば(1251)、同様にサーバ(16)へ転送する(1252)が、プリフェッチの準備は行わず、サーバ(16)からレスポンスを受信するのを待って(1253)それをクライアント(11)へ転送し(1254)、初期状態(1201)へ戻る。

【0089】

サーバ(16)へ転送したHTMLに対するGET(1204)のレスポンスが受信されるまで待機し(1205)受信されるとクライアント側プロキシ(15)へ転送する(1206)。同時にこのレスポンスを解析してファイル一覧を作成する(1207)。

【0090】

10

20

30

40

50

続いてクライアント側プロキシ(12)からのプリフェッチリクエストを待ち受ける(1208)。図11の1014及び1015がプリフェッチリクエストである。リクエストは連続して複数送られるため、全てのリクエストを受信するまでこれを繰り返す(1209)。続いてこれを順次サーバ(16)へ転送し(1210)、それに対するレスポンスを受信するまで待機する(1211)。少なくとも一のレスポンスを受信されれば、サーバ側プロキシ(15)は、そのレスポンスをクライアント側プロキシ(12)へ転送する(1212)。なお、図11の1023ないし1028のように、サーバ側プロキシは、一のファイルのリクエストに対するサーバからのレスポンス受領後に、別のファイルのリクエストを送信していてもよい。

【0091】

GET以外のリクエストを受信した場合(1255)は、同様にサーバへ転送する(1256)が、サーバ(16)からレスポンスを受信するのを待って(1257)それをクライアント側プロキシ(12)へ転送し(1258)、初期状態(1201)へ戻る。

【0092】

ファイル一覧にファイルが残っていれば、1210以降を繰り返す(1213)。すべてのデータを転送し終わるか、タイムアウトによりプリフェッチしたデータを破棄し初期状態(1201)に戻る(1214)。上記動作は、クライアント側プロキシ(12)、サーバ側プロキシ(15)の双方にてクライアント(11)およびサーバ(16)の組ごとに独立に行われる。クライアント(11)およびサーバ(16)の識別はパケットヘッダ中のIPアドレスを用いて行う。

【0093】

装置内の構成については実施例1の場合と同様であるので、ここでは省略する。

【実施例4】

【0094】

本発明の第四の実施例を図14に示す。この例では実施例1の複数のクライアントが、同一のフォルダをダウンロードした際に、プロキシにて読み込み済みのデータをバッファに保持し、一のクライアントからのアクセスで削除するのではなく、複数のクライアントからのアクセスのためにバッファに保持し、複数のクライアントにファイルを転送することにより、データ転送量の削減を図っている。

【0095】

この例では実施例1の場合にクライアントB(11b)が追加されており、クライアントA(11a)と同一のフォルダに対しダウンロードを試みている。クライアントA(11a)のシーケンスは実施例1と同様であるので、詳細は省略する。

【0096】

クライアント側プロキシ(12)では、フォルダカウント(1410)という値を用いてフォルダごとの同時アクセスしているクライアント数を管理する。クライアントA(11a)からのPROPFIND(201)を受信すると、クライアント側プロキシ(12)は、フォルダカウント(1410)を1に更新する。この値が1である間にクライアントB(11b)からPROPFIND(1401)が送信され、これを受信したクライアント側プロキシ(12)は、フォルダカウントを2に更新する(1411)。クライアント側プロキシ(12)は、PROPFINDに対するレスポンスを保持し、これをクライアントB(11b)に送信する(1402)。これを受けて、クライアント(11)がフォルダ内のファイルに対してGETを発行する(1403)。これに対するレスポンスもクライアント側プロキシ(12)が保持しているため、クライアント側プロキシ(12)はそれをそのまま送信する。(1404)。

【0097】

一方でクライアントA(11a)への転送はファイル3にて終了し(208)、クライアント側プロキシ(12)のフォルダカウントは1となる(1412)。

【0098】

クライアントB(11b)への転送はファイル2(1405、1406)、ファイル3(

10

20

30

40

50

1407、1408)と続き、終了する。このときクライアント側プロキシ(12)のフォルダカウントが1であるので、このカウンタおよびフォルダ内のファイルを削除して終了する。

【0099】

図15はクライアント側プロキシ(12)の内部構成を表す図である。本実施例では、実施例1の構成に共有データプール(1501)とフォルダテーブル(1502)を追加したものとなっている。レスポンスデータはクライアントごとのデータ管理テーブルではなく、クライアント(11a、11b)間で共有される共有データプール(1501)に格納される。また、フォルダテーブル(1502)はフォルダごとにそのフォルダにアクセスしているクライアント数を示すテーブルである。他は図3と同様である。

10

【0100】

図16はサーバ側プロキシ(15)の内部構成を表す図である。本実施例では、実施例1の構成に共有データプール(1601)とフォルダテーブル(1602)を追加したものとなっている。これらはクライアント側プロキシのものと同様の機能を持つ。他は図4と同様である。

【0101】

図17(a)はクライアント側プロキシ(12)のデータ管理部(304)における、データ管理テーブル(311~313)を示す。また図17(b)はサーバ側プロキシのデータ管理部(404)における、データ管理テーブル(411~413)を示す。実施例1との違いは、これらのテーブルではレスポンスを管理しないため、レスポンス内容(352、452)の項目がない点である。

20

【0102】

図17(c)は共有データプール(1501、1601)のデータテーブルを示す。これは、URL(1551)、レスポンス内容(1552)、ファイルが属するフォルダの情報(1553)を持つ。フォルダの情報は単なるフォルダ名でも、後述するフォルダテーブルのエントリへのポインタでよい。このデータはすべてのクライアント(11a、11b)で共有され、他のクライアントのリクエストによりプリフェッチされたデータであっても、データがこの中にあればこのデータをレスポンスとして送信する。

【0103】

図17(d)はフォルダテーブル(1502)を示す。これは、フォルダ名(1554)とカウンタ(1555)を持ち、カウンタ(1555)はフォルダごとにそのフォルダにアクセスしているクライアントの数を示しており、1以上の値をとる。フォルダ内全ファイルの転送が完了すると、プロキシはフォルダテーブル(1502)をチェックし、カウンタ(1555)の値が2以上であれば、他にアクセスしているクライアントがいるため同フォルダ内のファイルを削除しない。一方カウンタ(1555)の値が1であった場合は、他にアクセスしているクライアントがないのでファイルを削除する。

30

【0104】

図18はクライアント側プロキシ(12)の動作を表すフローチャートである。初期状態ではLAN側でパケットを受信するまで待機する(1801)。受信すると送信元および宛先IPアドレスごとに分離する(1802)。そのパケットがPROPFINDリクエストであれば(1803)フォルダテーブルをカウントアップする(1804)。その他のリクエストであれば(1851)、同様にサーバ側プロキシ(15)へ転送する(1852)が、サーバ側プロキシ(15)からレスポンスを受信するのを待って(1853)それをクライアント(11)へ転送し(1854)、初期状態(1801)へ戻る。

40

【0105】

フォルダカウントが2以上であれば(1805)、他のクライアントが読み込んだデータが存在するため、そのデータ(PROPFINDに対するレスポンス)をレスポンスとしてクライアント(11)へ送信する(1806)。フォルダカウントが1であれば、PROPFINDリクエストをサーバ側プロキシ(15)へ転送して(1807)GET待ち状態へ移行する。サーバ側プロキシ(15)へ転送したPROPFINDのレスポンスが受信されるまで待機し(1

50

808) 受信されるとクライアント(11)へ転送する(1809)。同時にこのレスポンスを解析してファイル一覧を作成する(1810)。

【0106】

続いてクライアントからのGETリクエストを待ち受ける(1811)。GET以外のリクエストを受信した場合(1855)は、同様にサーバ側プロキシ(15)へ転送する(556)が、WAN側からレスポンスを受信するのを待って(557)それをクライアント(11)へ転送し(558)、初期状態(501)へ戻る。

【0107】

GETを受信した場合、フォルダカウントを参照し(1812)、2以上であれば他のクライアントが読み込んだデータが存在するため、そのデータをレスポンスとしてクライアントへ送信する(1813)。フォルダ内の全ファイルを転送完了するかタイムアウトするまでこれを繰り返し(1814)、終了したらフォルダカウントをカウントダウンする(1815)。

【0108】

フォルダカウントが1であった場合、リクエストをサーバ側プロキシ(15)へ転送し(1816)、それに対するレスポンスを受信するまで待機して(1817)、受信されればクライアント(11)へ転送し(1817)、サーバ側プロキシ(15)からのプリフェッチデータを待ち受ける(1818)。サーバ側プロキシ(15)からのプリフェッチデータを受信すると(1818)、これをデータ管理部(304)に保存して(1820)、クライアントからのリクエストを待ち受ける(1821)。リクエストを受信すると保存していたデータをレスポンスとしてクライアントに送信する(1822)。

【0109】

ファイル一覧にファイルが残っていれば、512以降を繰り返す(1823)。すべてのデータを転送し終わるか、タイムアウトによりフォルダカウントをカウントダウンし、初期状態(1801)に戻る(1824)。

【0110】

図19はサーバ側プロキシ(15)の動作を表すフローチャートである。初期状態ではWAN側でパケットを受信するまで待機する(1901)。受信すると送信元および宛先IPアドレスごとに分離する(1902)。そのパケットがPROPFINDリクエストであれば(1903)フォルダテーブルをカウントアップする(1904)。その他のリクエストであれば(1951)、同様にサーバ(16)へ転送する(1952)が、サーバ(16)からレスポンスを受信するのを待って(1953)それをクライアント側プロキシ(12)へ転送し(1954)、初期状態(1901)へ戻る。

【0111】

フォルダカウントが2以上であれば(1905)、他のクライアントが読み込んだデータが存在するため、そのデータ(PROPFINDに対するレスポンス)をレスポンスとしてクライアント側プロキシ(12)へ送信する(1906)。フォルダカウントが1であれば、PROPFINDリクエストをサーバ(16)へ転送して(1907)GET待ち状態へ移行する。サーバ(16)へ転送したPROPFINDのレスポンスが受信されるまで待機し(1908)受信されるとクライアント側プロキシ(12)へ転送する(1909)。同時にこのレスポンスを解析してファイル一覧を作成する(1910)。

【0112】

続いてクライアント側プロキシ(12)からのGETリクエストを待ち受ける(1911)。GET以外のリクエストを受信した場合(1955)は、同様にサーバへ転送する(1956)が、サーバ(16)からレスポンスを受信するのを待って(1957)それをクライアント側プロキシ(12)へ転送し(1958)、初期状態(1901)へ戻る。

【0113】

GETを受信した場合、フォルダカウントを参照し(1912)、2以上であれば他のクライアントが読み込んだデータが存在するため、そのデータをレスポンスとしてクライアント側プロキシ(12)へ送信する(1913)。フォルダ内の全ファイルを転送完了する

10

20

30

40

50

がタイムアウトするまでこれを繰り返し（1914）、終了したらフォルダカウントをカウントダウンする（1915）。

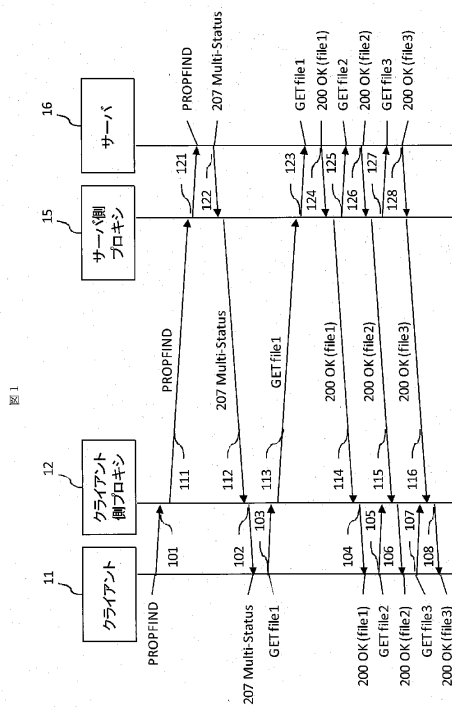
【0114】

フォルダカウントが1であった場合、リクエストをサーバ（16）へ転送し（1916）、それに対するレスポンスを受信するまで待機する（1917）。受信されればクライアント側プロキシ（12）へ転送し（1918）、次のファイルに対するプリフェッチリクエストを送信する（1919）。サーバ（16）からのプリフェッチデータを受信すると（1920）、クライアント側プロキシに送信する（1921）。ファイル一覧にファイルが残っていれば、1919以降を繰り返す（1922）。すべてのデータを転送し終わるか、タイムアウトによりプリフェッチしたデータを破棄し初期状態（1901）に戻る（1923）。

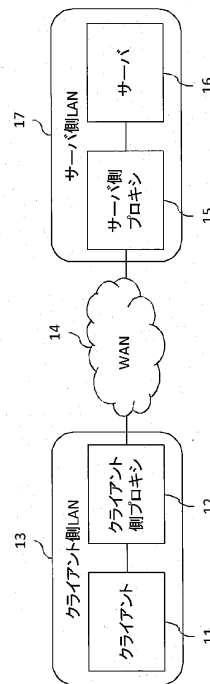
【0115】

上述の実施例により、HTTPやWebDAVに従う通信で、サーバからクライアントの利用状況に応じて複数ファイルを効率的に転送することができる。

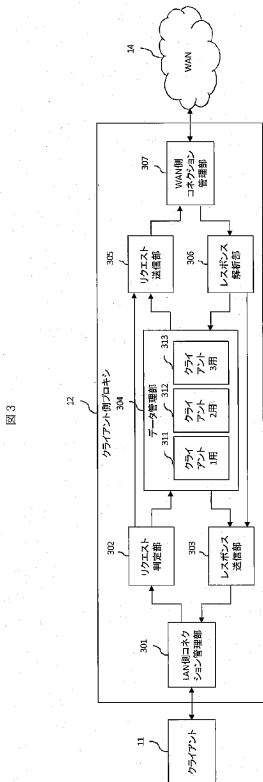
【図1】



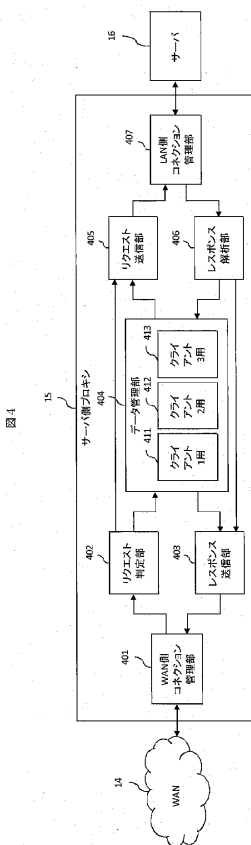
【図2】



【図3】



【図4】



【図5】

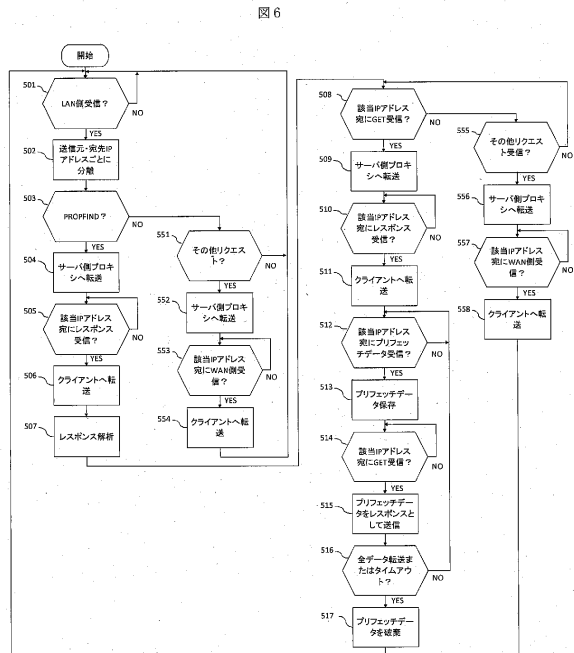
図5

311		353		354		355	
URL	レスポンス内容	クライアントからリクエスト受信済み	サーバ側プロキシからレスポンス受信済み	クライアントにレスポンス送信済み	URL	レスポンス内容	サーバ側プロキシからリクエスト受信済み
file1	200 OK ...	1	1	1	file1	200 OK ...	1
file2	200 OK ...	0	0	0	file2	200 OK ...	0
file3		0	0	0	file3		0

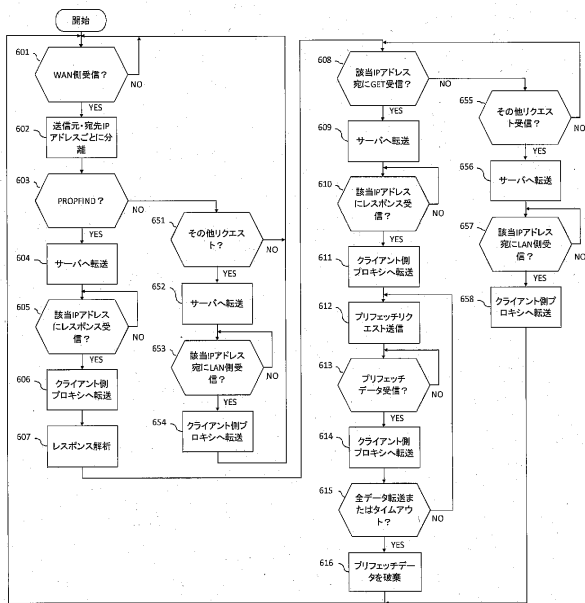
411		453		454		455	
URL	レスポンス内容	クライアント側プロキシからリクエスト受信済み	サーバ側プロキシからリクエスト受信済み	サーバ側プロキシからレスポンス送信済み	URL	レスポンス内容	サーバ側プロキシからリクエスト受信済み
file1	200 OK ...	1	0	1	file1	200 OK ...	1
file2		0	0	0	file2		0
file3		0	0	0	file3		0

【図6】



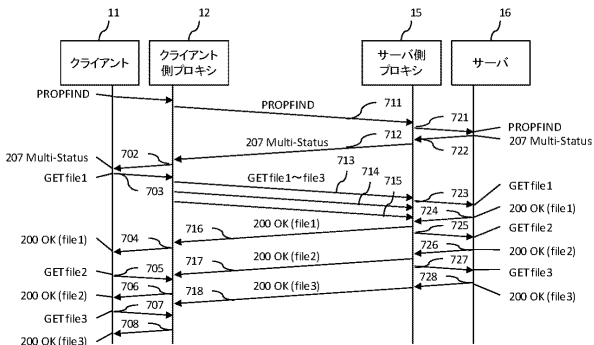
【図7】

図7



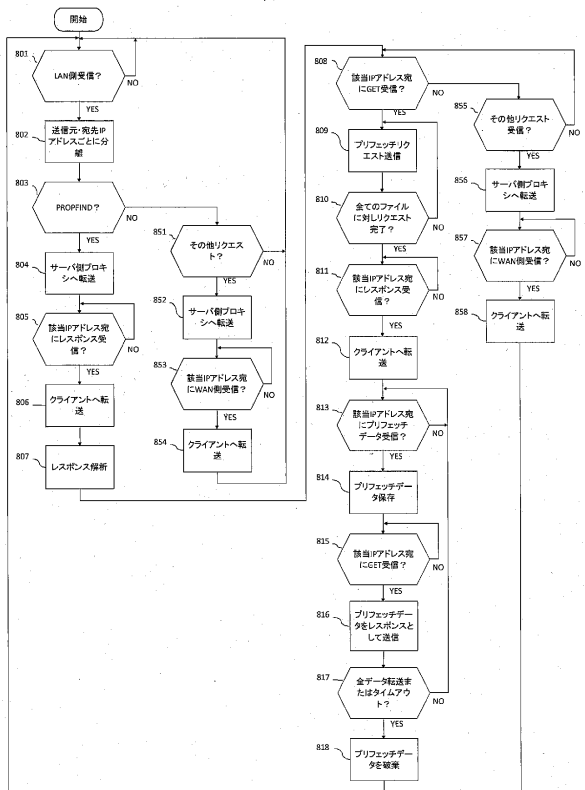
【図8】

【図8】



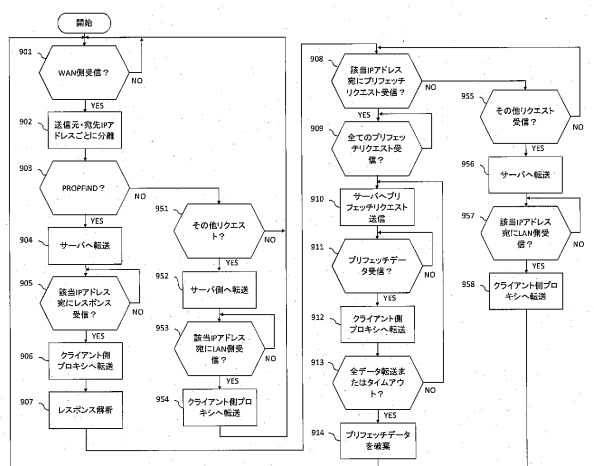
【図9】

図9

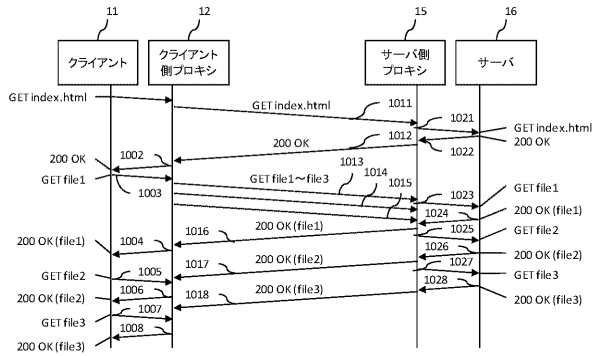


【図10】

図10



【図11】



【図11】

【図12】

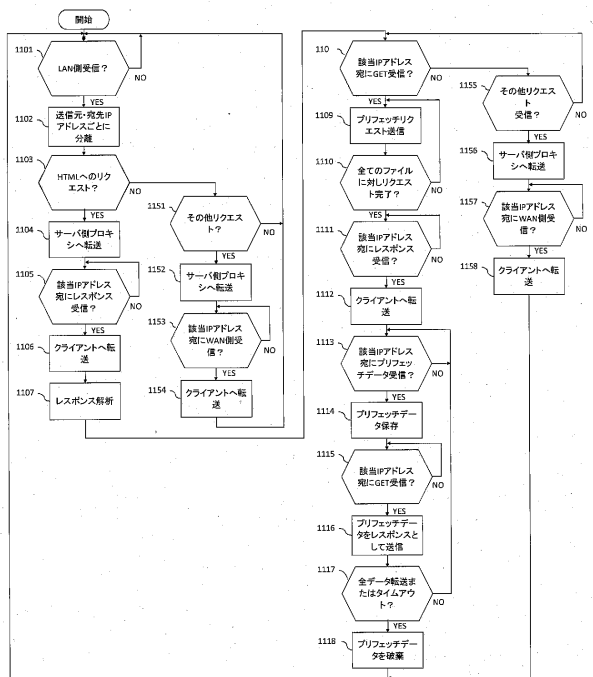


図12

【図13】

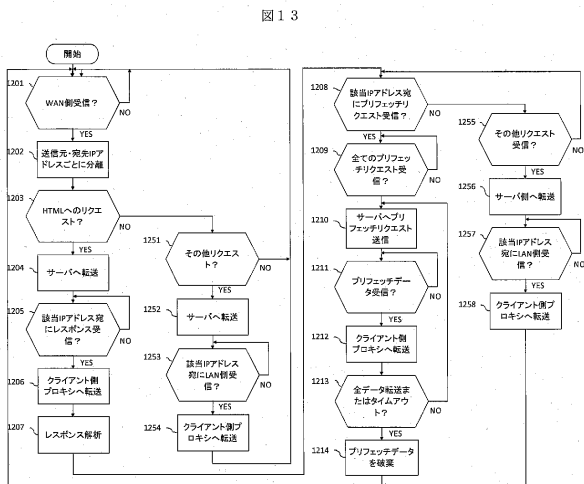


図13

【図14】

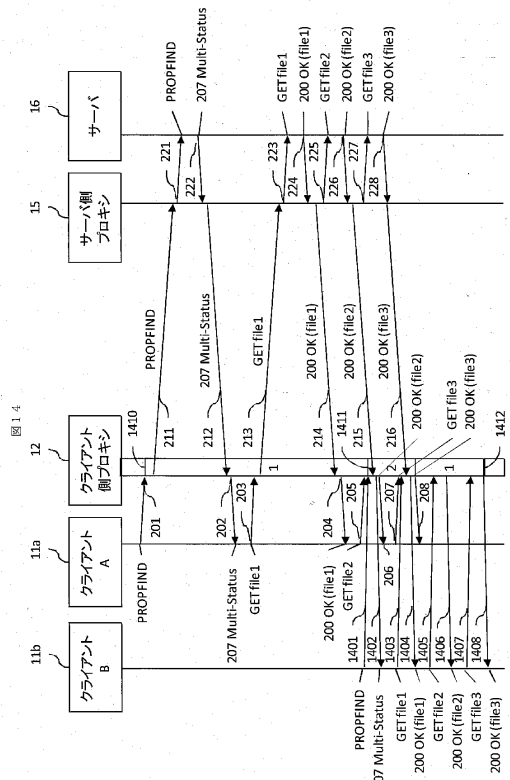
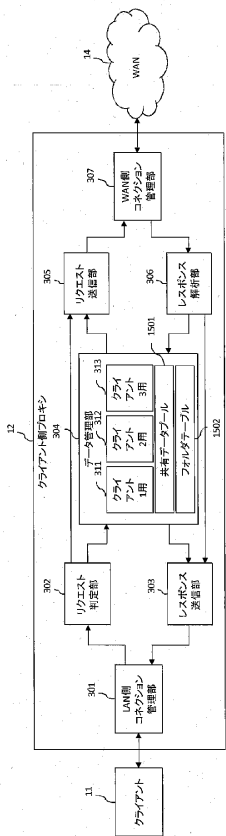


図14

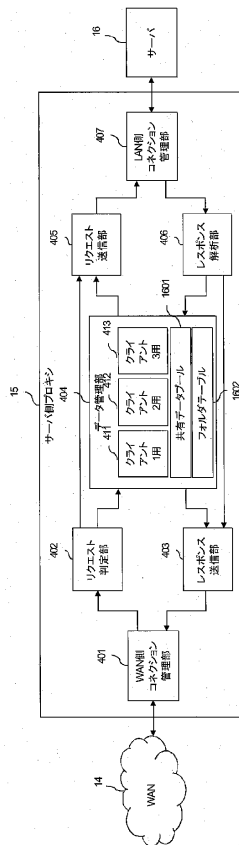
【図15】

図15



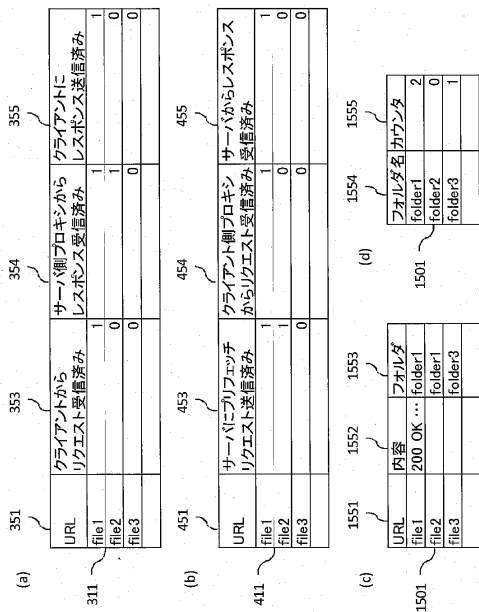
【図16】

図16



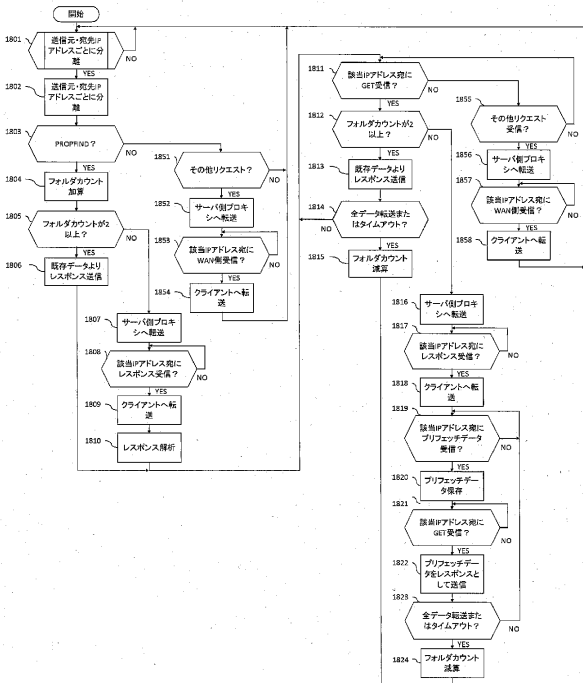
【図17】

図17



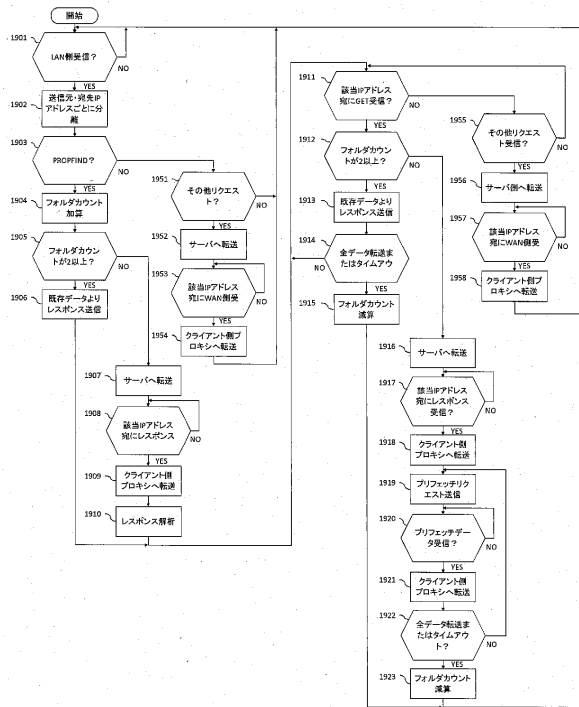
【図18】

図18



【 図 19 】

図 19



## フロントページの続き

(72)発明者 明石 大

東京都国分寺市東恋ヶ窪一丁目280番地 株式会社日立製作所中央研究所内

(72)発明者 伊藤 大輔

東京都国分寺市東恋ヶ窪一丁目280番地 株式会社日立製作所中央研究所内

審査官 古河 雅輝

(56)参考文献 特開2012-003652(JP,A)

特開2010-033112(JP,A)

国際公開第2008/099897(WO,A1)

特開2003-186785(JP,A)

特開2001-209571(JP,A)

(58)調査した分野(Int.Cl., DB名)

G06F 12/00

G06F 12/08 - 12/12

G06F 13/00

H04L 12/00 - 12/26

H04L 12/50 - 12/955