

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第6503988号
(P6503988)

(45) 発行日 平成31年4月24日(2019.4.24)

(24) 登録日 平成31年4月5日(2019.4.5)

(51) Int. Cl.		F I			
G06F	13/00	(2006.01)	G06F	13/00	353C
H04L	12/70	(2013.01)	H04L	12/70	A
H04L	29/08	(2006.01)	H04L	13/00	307A

請求項の数 5 (全 37 頁)

(21) 出願番号	特願2015-172754 (P2015-172754)	(73) 特許権者	000005223 富士通株式会社 神奈川県川崎市中原区上小田中4丁目1番1号
(22) 出願日	平成27年9月2日(2015.9.2)	(74) 代理人	100092152 弁理士 服部 毅巖
(65) 公開番号	特開2017-49819 (P2017-49819A)	(72) 発明者	久保田 真 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内
(43) 公開日	平成29年3月9日(2017.3.9)	審査官	木村 雅也
審査請求日	平成30年5月14日(2018.5.14)		

最終頁に続く

(54) 【発明の名称】 セッション制御方法およびセッション制御プログラム

(57) 【特許請求の範囲】

【請求項1】

第1の装置と第2の装置とが中継プロセスを介して通信可能であり、第3の装置と前記第2の装置とが前記中継プロセスを介して通信可能であるシステムが実行する、セッション制御方法であって、

前記中継プロセスの前記第1の装置側に確立された第1のコネクションと、前記中継プロセスの前記第2の装置側に確立された第2のコネクションとを対応付けた制御情報を記憶し、

前記第1の装置から前記第2の装置に対して、前記第1のコネクションおよび前記第2のコネクションより上位の通信レイヤの切断指示メッセージが発行された場合、前記中継プロセスにおいて前記切断指示メッセージを終端し、また、前記切断指示メッセージの発行に応じて前記制御情報を更新し、

前記第1の装置から前記第3の装置への切り替えが通知され、かつ、前記切断指示メッセージが発行されたことを前記制御情報が示している場合、前記中継プロセスの前記第3の装置側に確立された第3のコネクションを使用可能にし、前記第2のコネクションと前記第3のコネクションとが対応付けられるように前記制御情報を更新する、

セッション制御方法。

【請求項2】

前記切断指示メッセージが、前記切り替えが通知される前であって前記第1の装置から前記第3の装置への切り替えの開始が通知された後に発行された場合には、前記中継プロ

セスにおいて前記切断指示メッセージを端末し、

前記切断指示メッセージが、前記切り替えの開始が通知される前に発行された場合には、前記第2の装置に前記切断指示メッセージを転送する、

請求項1記載のセッション制御方法。

【請求項3】

前記切り替えが通知される前に、前記中継プロセスと前記第3の装置との間に前記第3のコネクションを確立する、

請求項1記載のセッション制御方法。

【請求項4】

前記中継プロセスが実行される装置の負荷に応じて、前記切り替えが通知される前に前記第3のコネクションを確立するか、前記切り替えが通知された後に前記第3のコネクションを確立するかを選択する、

請求項1記載のセッション制御方法。

【請求項5】

コンピュータに、

第1の装置側に確立された第1のコネクションと、第2の装置側に確立された第2のコネクションとを対応付けた制御情報を記憶し、

前記第1の装置から、前記第1のコネクションおよび前記第2のコネクションより上位の通信レイヤのメッセージであって前記第2の装置宛ての切断指示メッセージを受信した場合、前記切断指示メッセージを端末し、また、前記切断指示メッセージの受信に応じて前記制御情報を更新し、

前記第1の装置から第3の装置への切り替えが通知され、かつ、前記切断指示メッセージを受信したことを前記制御情報が示している場合、前記第3の装置側に確立された第3のコネクションを使用可能にし、前記第2のコネクションと前記第3のコネクションとが対応付けられるように前記制御情報を更新する、

処理を実行させるセッション制御プログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明はセッション制御方法およびセッション制御プログラムに関する。

【背景技術】

【0002】

従来、クライアントとサーバ間の通信には、コネクションを長時間維持しないステートレスな通信プロトコルを用いることが多かった。例えば、HTTP (Hypertext Transfer Protocol) の場合、クライアントがHTTPメッセージを送信したいときにサーバとの間でTCP (Transmission Control Protocol) コネクションを確立し、ひとまとまりのHTTPメッセージの送信が完了するとTCPコネクションを切断する。ステートレス型の通信プロトコルは、クライアントとサーバ間で頻繁にデータを送信する場合や、双方向にリアルタイムにデータを送信する場合に、通信コストが大きくなりやすい。

【0003】

これに対し、クライアントとサーバ間の通信に、コネクションを長時間維持するステートフルな通信プロトコルを用いることもできる。例えば、IETF (Internet Engineering Task Force) のRFC (Request for Comments) 6455に、Webソケットの仕様が定義されている。Webソケットでは、クライアントとサーバがTCPコネクションを確立し、TCPコネクション上でWebソケットメッセージを相互に送信する。確立されたTCPコネクションは、ひとまとまりのWebソケットメッセージの送信が完了してもすぐに切断しなくてよく、クライアントまたはサーバが次のWebソケットメッセージを送信する際に引き続き使用することができる。

【0004】

クライアントとサーバが通信可能な状態を維持しなくてよくなったときは、一方から他

10

20

30

40

50

方に対してクローズを示すWebソケットメッセージ(クローズメッセージ)を送信する。クローズメッセージが送信されると、Webソケットより下位の通信レイヤの接続であるTCP接続は不要になるため、TCP接続が切断される。ステートフル型の通信プロトコルは、クライアントとサーバの間で頻繁にデータを送信する場合や、双方向にリアルタイムにデータを送信する場合に、ステートレス型の通信プロトコルと比べて通信コストを小さく抑えることができる。

【0005】

なお、第2の端末と通信する端末を、通信中に第1の端末から第3の端末に切り替えることができるセッション制御方法が提案されている。提案のセッション制御方法では、第1の端末で実行されるローカルプロキシが、第1の端末と第2の端末の間のセッションについて10の制御情報を保存しておく。ローカルプロキシは、ユーザから端末の切り替えが指示されると、制御情報に基づいて、第2の端末に第3の端末とセッションを確立するよう要求する。ローカルプロキシは、セッション確立が完了した旨の通知を第2の端末から受信すると、第1の端末と第2の端末の間のセッションを切断する。

【0006】

また、クライアントが送信するリクエストメッセージを複数のサーバに振り分ける切替装置(ロードバランサ)が提案されている。提案の切替装置は、クライアントとの間にTCP接続を確立し、クライアントからリクエストメッセージを受信する。切替装置は、複数のサーバのうちの1つを選択し、選択したサーバとの間にTCP接続を確立し、選択したサーバにリクエストメッセージを転送する。切替装置は、選択したサーバからFINパケットを受信すると、選択したサーバとの間のTCP接続を切断する。また、切替装置は、FINパケットをクライアントに転送し、クライアントとの間のTCP接続を切断する。20

【0007】

また、コンテンツを再生する端末を第1の端末から第2の端末に切り替えることができるシステムが提案されている。提案のシステムでは、第1の端末がコンテンツサーバとの間にセッションを確立し、コンテンツサーバからコンテンツの受信を開始する。端末を切り替える場合、第1の端末は、コンテンツサーバにコンテンツ配信の停止を要求し、アプリケーションサーバにセッションIDを通知する。第2の端末は、第1の端末が使用していたセッションIDをアプリケーションサーバから取得し、取得したセッションIDを指定してコンテンツサーバにコンテンツ配信の再開を要求する。30

【0008】

また、クライアントからの要求に応じて、複数のサーバを用いてクライアントにサービスを提供するロードバランサが提案されている。提案のロードバランサは、クライアントとの間にTCP接続を確立し、クライアントからサービス要求を受信する。ロードバランサは、TCP接続の識別情報を含む「インタレスト」を生成し、複数のサーバに同報送信する。ロードバランサは、インタレストに対する応答が最も早いサーバを選択し、選択したサーバとクライアントとの間にセッションが確立するよう制御する。

【0009】

また、SSL(Secure Socket Layer)によってカプセル化されたパケットを、使用されている上位プロトコルに応じてフィルタリングする中継装置が提案されている。提案の中継装置は、複数の上位プロトコルそれぞれについて宛先のサーバと通信を試行し、通信が成功した上位プロトコルを判定する。中継装置は、通信が成功した上位プロトコルを、クライアントが使用しようとしている上位プロトコルであると推定し、推定した上位プロトコルに応じてパケットを中継するか破棄するか決定する。40

【先行技術文献】**【特許文献】****【0010】**

【特許文献1】特開2005-32172号公報

【特許文献2】国際公開第2006/126403号

10

20

30

40

50

【特許文献3】国際公開第2010/049863号

【特許文献4】特開2012-38299号公報

【特許文献5】特開2012-147415号公報

【発明の概要】

【発明が解決しようとする課題】

【0011】

ところで、2つの装置の間に接続が維持されているときに、負荷や保守などの都合で一方の装置を切り替えたい場合がある。例えば、クライアントとサーバとがWebソケット通信のためにTCP接続を維持しているときに、クライアントと通信するサーバを切り替えたい場合がある。一方の装置を切り替えると、通信アドレスが変わってしまうため、切り替え前の接続をそのまま使用し続けることができない。このため、他方の装置は接続が切断されて通信不可になり、切り替え後の装置と接続を確立し直すことになる。このとき、切り替え後の装置へのアクセスにユーザの操作を要することがあり、ユーザの利便性が低下するおそれがある。

10

【0012】

一方の装置の切り替えの影響を軽減する方法としては、2つの装置の通信経路上でリバースプロキシなどの中継プロセスを実行する方法が考えられる。中継プロセスは、一方の装置と接続を確立し、これとは別に他方の装置と接続を確立し、2つの接続の間でメッセージを転送する。

【0013】

しかし、中継プロセスは通常、上位の通信レイヤのメッセージの内容に関係なく、一方の接続から受信したメッセージを他方の接続に転送するに過ぎない。このため、一方の装置から他方の装置に対して、上位の通信レイヤの切断指示メッセージが発行されると、その切断指示メッセージも中継プロセスを超えて送信されてしまう。その結果、切断指示メッセージを受けて他方の装置が接続を切断してしまうという問題がある。例えば、リバースプロキシは通常、Webソケットメッセージをそのまま転送するため、サーバからクライアントに対するクローズメッセージも転送する。クローズメッセージを受けて、クライアントはTCP接続を切断してしまう。

20

【0014】

すなわち、中継プロセスを実行することで2つの装置の間の接続を分割しても、上位の通信レイヤの指示によって両方の接続が切断されてしまうことがある。

30

1つの側面では、本発明は、通信相手を切り替える際の接続の切断を抑制できるセッション制御方法およびセッション制御プログラムを提供することを目的とする。

【課題を解決するための手段】

【0015】

1つの態様では、第1の装置と第2の装置とが中継プロセスを介して通信可能であり、第3の装置と第2の装置とが中継プロセスを介して通信可能であるシステムが実行する、セッション制御方法が提供される。中継プロセスの第1の装置側に確立された第1の接続と、中継プロセスの第2の装置側に確立された第2の接続とを対応付けた制御情報を記憶する。第1の装置から第2の装置に対して、第1の接続および第2の接続より上位の通信レイヤの切断指示メッセージが発行された場合、中継プロセスにおいて切断指示メッセージを終端し、また、切断指示メッセージの発行に応じて制御情報を更新する。第1の装置から第3の装置への切り替えが通知され、かつ、切断指示メッセージが発行されたことを制御情報が示している場合、中継プロセスの第3の装置側に確立された第3の接続を使用可能にし、第2の接続と第3の接続とが対応付けられるように制御情報を更新する。

40

【0016】

また、1つの態様では、セッション制御プログラムが提供される。

【発明の効果】

【0017】

50

1つの側面では、通信相手を切り替える際のコネクションの切断を抑制できる。

【図面の簡単な説明】

【0018】

【図1】第1の実施の形態の情報処理システムの例を示す図である。

【図2】第2の実施の形態の情報処理システムの例を示す図である。

【図3】Webサーバのハードウェア例を示すブロック図である。

【図4】第2の実施の形態のサーバアプリケーションの配備例を示す図である。

【図5】HTTPパケットとWebソケットパケットの例を示す図である。

【図6】Webソケットによる第1の通信例を示すシーケンス図である。

【図7】Webソケットによる第2の通信例を示すシーケンス図である。

10

【図8】Webソケットによる第3の通信例を示すシーケンス図である。

【図9】第2の実施の形態の移動制御の通信例を示すシーケンス図である。

【図10】配備計画サーバの状態遷移例を示す図である。

【図11】第2の実施の形態のリバースプロキシの機能例を示すブロック図である。

【図12】リバースプロキシが有する制御テーブルの例を示す図である。

【図13】配備計画サーバが有する制御テーブルの例を示す図である。

【図14】上り通信転送の手順例を示すフローチャートである。

【図15】下り通信転送の手順例を示すフローチャートである。

【図16】プロキシ状態制御の手順例を示すフローチャートである。

【図17】第3の実施の形態のサーバアプリケーションの配備例を示す図である。

20

【図18】Webソケットによる第4の通信例を示すシーケンス図である。

【図19】Webソケットによる第4の通信例を示すシーケンス図(続き)である。

【図20】第3の実施の形態の移動制御の通信例を示すシーケンス図である。

【図21】第3の実施の形態のリバースプロキシの機能例を示すブロック図である。

【発明を実施するための形態】

【0019】

以下、本実施の形態を図面を参照して説明する。

[第1の実施の形態]

図1は、第1の実施の形態の情報処理システムの例を示す図である。

【0020】

30

第1の実施の形態の情報処理システムは、中継プロセス10、装置21(第1の装置)、装置22(第2の装置)および装置23(第3の装置)を有する。装置21~23は、情報処理装置やコンピュータと呼ばれるものであってもよい。装置21~23は、ユーザが操作する端末装置(クライアントコンピュータなど)でもよいし、端末装置からアクセスされるサーバ装置(サーバコンピュータなど)でもよい。一例として、装置22が端末装置であり、装置21,23がサーバ装置である形態が考えられる。

【0021】

装置21と装置22とは、中継プロセス10を介して通信可能である。また、装置23と装置22とは、中継プロセス10を介して通信可能である。中継プロセス10は、装置21,23と装置22との間でデータを転送する。中継プロセス10は、リバースプロキシであってよい。例えば、中継プロセス10は、装置22から具体的な通信相手(装置21または装置23)を隠蔽する。中継プロセス10は、装置21に配置されてもよいし、装置23に配置されてもよいし、装置21~23と異なる装置に配置されてもよい。

40

【0022】

ここで、装置22の通信相手が最初は装置21であり、装置22の通信相手を途中で装置21から装置23に切り替える場合を考える。装置21から装置23への切り替えは、例えば、装置21の負荷が高い場合、中継プロセス10と装置21の間の通信路が輻輳している場合、装置21の保守作業を行う場合などに実行され得る。このとき、装置22と通信するアプリケーションプロセスを、装置21から装置23に移動してもよい。

【0023】

50

装置 2 2 の通信相手が装置 2 1 であるとき、中継プロセス 1 0 は、装置 2 1 側に接続 1 1 を確立し、装置 2 2 側に接続 1 2 を確立する。接続 1 1 , 1 2 は、例えば、TCP 接続である。中継プロセス 1 0 が装置 2 1 , 2 2 と異なる他の装置（装置 2 3 でもよい）に配置されている場合、例えば、当該他の装置と装置 2 1 との間に接続 1 1 が確立され、当該他の装置と装置 2 2 との間に接続 1 2 が確立される。中継プロセス 1 0 が装置 2 1 に配置されている場合、中継プロセス 1 0 の装置 2 1 側の接続 1 1 は、装置 2 1 に配置されたアプリケーションプロセスと中継プロセス 1 0 との間の同一装置内の接続を意味する。

【 0 0 2 4 】

中継プロセス 1 0 は、接続 1 1 と接続 1 2 とを対応付けた制御情報 1 5 を記憶する。制御情報 1 5 は、例えば、中継プロセス 1 0 が配置された装置が有する記憶装置に記憶される。記憶装置は、RAM (Random Access Memory) などの揮発性の記憶装置でもよいし、HDD (Hard Disk Drive) などの不揮発性の記憶装置でもよい。

【 0 0 2 5 】

中継プロセス 1 0 は、制御情報 1 5 に基づいて、接続 1 1 と接続 1 2 との間でデータを転送する。すなわち、中継プロセス 1 0 は、接続 1 1 から受信したデータを接続 1 2 に送信し、接続 1 2 から受信したデータを接続 1 1 に送信する。接続 1 1 , 1 2 上で送信されるデータには、接続 1 1 , 1 2 より上位の通信レイヤのメッセージが含まれる。例えば、接続 1 1 , 1 2 が TCP 接続である場合、TCP より上位の通信プロトコルである Web 20
ソケットのメッセージが、接続 1 1 , 1 2 上で送信され得る。

【 0 0 2 6 】

装置 2 2 の通信相手を装置 2 1 から装置 2 3 に切り替える場合、通信アドレス（例えば、IP (Internet Protocol) アドレスや TCP ポート番号）が変わるため、接続 1 1 を装置 2 3 に引き継ぐことはできない。このとき、装置 2 1 が装置 2 2 に対して、接続 1 1 , 1 2 より上位の通信レイヤの切断指示メッセージ 1 4 を送信することがある。接続 1 1 , 1 2 が TCP 接続である場合、切断指示メッセージ 1 4 は、TCP の制御パケットである FIN パケットとは異なる。切断指示メッセージ 1 4 の例として、Web ソケットのクローズメッセージなどが挙げられる。

【 0 0 2 7 】

中継プロセス 1 0 は、接続 1 1 から切断指示メッセージ 1 4 を受信すると、切断指示メッセージ 1 4 を終端する。すなわち、中継プロセス 1 0 は、切断指示メッセージ 1 4 を、指定された宛先（装置 2 2 ）に転送せずに破棄する。切断指示メッセージ 1 4 が装置 2 2 に転送されると、装置 2 2 は、上位の通信レイヤのサービスが終了または中断されたと認識することになり、接続 1 2 を切断してしまう。一方、切断指示メッセージ 1 4 が中継プロセス 1 0 でブロックされると、装置 2 2 は、上位の通信レイヤのサービスが継続していると認識し、接続 1 2 を切断しない。

【 0 0 2 8 】

ただし、中継プロセス 1 0 は、装置 2 2 の通信相手を切り替えようとしている場合のみ切断指示メッセージ 1 4 を終端し、それ以外の場合は切断指示メッセージ 1 4 を装置 2 2 40
に転送するようにしてもよい。通信相手を切り替えようとしているか否かは、このシステムを管理する管理サーバが中継プロセス 1 0 に通知してもよい。また、中継プロセス 1 0 は、切断指示メッセージ 1 4 を受信すると、制御情報 1 5 を更新して、切断指示メッセージ 1 4 を受信したことを制御情報 1 5 に反映させる。例えば、中継プロセス 1 0 は、切断指示メッセージ 1 4 を受信したことを示す情報を、接続 1 1 と対応付けて制御情報 1 5 に記録する。また、例えば、中継プロセス 1 0 は、接続 1 1 が不要になったことまたは接続 1 1 を切断したことを、制御情報 1 5 に記録する。

【 0 0 2 9 】

その後、中継プロセス 1 0 は、装置 2 1 から装置 2 3 への切り替えの通知を受け付ける。切り替えの通知は、例えば、このシステムを管理する管理サーバが中継プロセス 1 0 に 50

通知する。切り替えの通知は、装置 2 1 から装置 2 3 へのアプリケーションプロセスの移動が完了したことを示す通知であってもよい。中継プロセス 1 0 は、切り替えの通知があり、かつ、制御情報 1 5 が切断指示メッセージ 1 4 を既に受信したことを示している場合、装置 2 3 側に確立されたコネクション 1 3 を使用可能にする。

【 0 0 3 0 】

コネクション 1 3 は、コネクション 1 1 , 1 2 と同じ種類のコネクションであり、例えば、TCPコネクションである。中継プロセス 1 0 が装置 2 3 と異なる他の装置（装置 2 1 でもよい）に配置されている場合、例えば、当該他の装置と装置 2 3 との間にコネクション 1 3 が確立される。中継プロセス 1 0 が装置 2 3 に配置されている場合、中継プロセス 1 0 の装置 2 3 側のコネクション 1 3 は、装置 2 3 に配置されたアプリケーションプロセスと中継プロセス 1 0 との間の同一装置内のコネクションを意味する。

10

【 0 0 3 1 】

中継プロセス 1 0 は、切り替えの通知を受け付けてからコネクション 1 3 を確立してもよいし、切り替えの通知を受け付ける前に予めコネクション 1 3 を確立しておいてもよい。前者の場合、コネクション 1 3 を使用可能にすることは、コネクション 1 3 を確立することを含んでもよい。後者の場合、コネクション 1 3 を使用可能にすることは、装置 2 2 にコネクション 1 3 を割り当てることを含んでもよい。また、コネクション 1 3 を使用可能にすることは、コネクション 1 3 に上位の通信レイヤのサービスを対応付ける（例えば、コネクション 1 3 上で上位の通信レイヤのハンドシェイクを行う）ことを含んでもよい。

20

【 0 0 3 2 】

中継プロセス 1 0 は、コネクション 1 3 を使用可能にすると、コネクション 1 2 とコネクション 1 3 とを対応付け、その対応関係を示すように制御情報 1 5 を更新する。例えば、コネクション 1 1 とコネクション 1 2 の対応関係に代えて、コネクション 1 2 とコネクション 1 3 の対応関係が制御情報 1 5 に登録される。以降、中継プロセス 1 0 は、制御情報 1 5 に基づいて、コネクション 1 2 とコネクション 1 3 との間でデータを転送する。すなわち、中継プロセス 1 0 は、コネクション 1 2 から受信したデータをコネクション 1 3 に送信し、コネクション 1 3 から受信したデータをコネクション 1 2 に送信する。

【 0 0 3 3 】

第 1 の実施の形態の情報処理システムによれば、装置 2 2 の通信相手を装置 2 1 から装置 2 3 に切り替えるとき、装置 2 1 から装置 2 2 に対して発行された上位の通信レイヤの切断指示メッセージ 1 4 が、中継プロセス 1 0 において終端される。これにより、上位の通信レイヤのサービスが継続していると装置 2 2 に認識させることができ、切断指示メッセージ 1 4 に反応して装置 2 2 がコネクション 1 2 を切断してしまうことを抑制できる。

30

【 0 0 3 4 】

そして、切り替え完了後は、コネクション 1 2 とコネクション 1 3 が対応付けられ、中継プロセス 1 0 によって装置 2 2 と装置 2 3 との間でデータが中継される。これにより、装置 2 2 は、切り替え前に確立したコネクション 1 2 を切り替え後も使用し続けることができ、新たにコネクションを確立し直さなくてよい。よって、装置 2 2 のユーザは、ログイン手続などの操作を行わずに切り替え後もサービスを継続して利用することができ、ユーザの利便性が低下しないようにすることができる。

40

【 0 0 3 5 】

[第 2 の実施の形態]

次に、第 2 の実施の形態を説明する。

図 2 は、第 2 の実施の形態の情報処理システムの例を示す図である。

【 0 0 3 6 】

第 2 の実施の形態の情報処理システムは、クラウド 3 0、拠点 4 0, 5 0 およびネットワーク 6 0 を含む。クラウド 3 0 および拠点 4 0, 5 0 は、互いに離れた場所に設置されており、広域のネットワーク 6 0 を介して接続されている。

【 0 0 3 7 】

50

クラウド30は、複数の拠点から利用できる共有の情報処理施設であり、例えば、データセンタを用いて実現される。拠点40, 50は、組織のメンバーが業務を行う施設であり、例えば、会社の従業員が業務を行うオフィスや工場である。

【0038】

クラウド30は、Webサーバ100aおよび配備計画サーバ200を含む。拠点40は、端末装置41, 42、DNS (Domain Name System) サーバ43およびWebサーバ100を含む。端末装置41, 42、DNSサーバ43およびWebサーバ100は、拠点40内のローカルのネットワーク44に接続されている。ネットワーク44は、ネットワーク60に接続されている。拠点50は、端末装置51, 52、DNSサーバ53およびWebサーバ100bを含む。端末装置51, 52、DNSサーバ53およびWebサーバ100bは、拠点50内のローカルのネットワーク54に接続されている。ネットワーク54は、ネットワーク60に接続されている。

10

【0039】

端末装置41, 42, 51, 52は、ユーザが操作するクライアントコンピュータである。端末装置41, 42, 51, 52は、Webブラウザなどのクライアントアプリケーションを実行し、Webサーバ100, 100a, 100bで実行されるサーバアプリケーションと通信する。DNSサーバ43, 53は、ドメイン名とIPアドレスとの対応関係を管理するサーバコンピュータである。DNSサーバ43は、端末装置41, 42からドメイン名を指定したアドレス解決要求を受信し、ドメイン名に対応するIPアドレスを回答する。同様に、DNSサーバ53は、端末装置51, 52からドメイン名を指定した

20

【0040】

Webサーバ100, 100a, 100bは、クライアントアプリケーションに対してサービスを提供するサーバアプリケーションを実行することができる。サーバアプリケーションは、URLによって識別される。第2の実施の形態では、サーバアプリケーションは、Webサーバ100, 100a, 100bの間で移動することができる。端末装置41, 42, 51, 52がサーバアプリケーションの位置を認識しなくてよいように、Webサーバ100, 100bは更にリバースプロキシを実行する。

【0041】

サーバアプリケーションに対しては、当該サーバアプリケーションが配置されたWebサーバに依存する実URLとは別に、Webサーバに依存しない仮想URLが付与される。DNSサーバ43は、仮想URLに対してWebサーバ100のIPアドレスを対応付ける。これにより、端末装置41, 42のユーザが仮想URLを指定すると、クライアントアプリケーションはWebサーバ100にアクセスすることになる。Webサーバ100のリバースプロキシは、仮想URLに対応するサーバアプリケーションがWebサーバ100に存在しない場合、適切なWebサーバにアクセスを転送する。

30

【0042】

同様に、DNSサーバ53は、仮想URLに対してWebサーバ100bのIPアドレスを対応付ける。これにより、端末装置51, 52のユーザが仮想URLを指定すると、クライアントアプリケーションはWebサーバ100bにアクセスすることになる。Webサーバ100bのリバースプロキシは、仮想URLに対応するサーバアプリケーションがWebサーバ100bに存在しない場合、適切なWebサーバにアクセスを転送する。これにより、端末装置41, 42, 51, 52のクライアントアプリケーションは、サーバアプリケーションの位置を意識せずにサーバアプリケーションと通信できる。

40

【0043】

配備計画サーバ200は、Webサーバ100, 100a, 100bへのサーバアプリケーションの配置を制御するサーバコンピュータである。サーバアプリケーションの配置は、クラウド30と拠点40, 50との間の通信状況、Webサーバ100, 100a, 100bの負荷、Webサーバ100, 100a, 100bの保守スケジュールなどに応じて決定される。例えば、最初は、クラウド30内のWebサーバ100aにサーバアプ

50

リケーションが配置される。拠点40から当該サーバアプリケーションへのアクセスが多く、ネットワーク60の遅延が大きい場合、Webサーバ100aからWebサーバ100にサーバアプリケーションを移動することが考えられる。

【0044】

なお、サーバアプリケーションの配置は、サーバアプリケーション毎、すなわち、仮想URL毎に変更することが可能である。また、図2では配備計画サーバ200をクラウド30に設置したが、拠点40または拠点50に設置することもできる。

【0045】

図3は、Webサーバのハードウェア例を示すブロック図である。

Webサーバ100は、CPU101、RAM102、HDD103、画像信号処理部104、入力信号処理部105、媒体リーダ106および通信インタフェース107を有する。CPU101、RAM102、HDD103、画像信号処理部104、入力信号処理部105、媒体リーダ106および通信インタフェース107は、バス108に接続されている。なお、端末装置41, 42, 51, 52、DNSサーバ43, 53、Webサーバ100a, 100bおよび配備計画サーバ200も、Webサーバ100と同様のハードウェアを用いて実現することができる。

【0046】

CPU101は、プログラムの命令を実行する演算回路を含むプロセッサである。CPU101は、HDD103に記憶されたプログラムやデータの少なくとも一部をRAM102にロードし、プログラムを実行する。なお、CPU101は複数のプロセッサコアを備えてもよく、Webサーバ100は複数のプロセッサを備えてもよく、以下で説明する処理を複数のプロセッサまたはプロセッサコアを用いて並列に実行してもよい。また、複数のプロセッサの集合(マルチプロセッサ)を「プロセッサ」と呼んでもよい。

【0047】

RAM102は、CPU101が実行するプログラムやCPU101が演算に用いるデータを一時的に記憶する揮発性の半導体メモリである。なお、Webサーバ100は、RAM以外の種類のメモリを備えてもよく、複数個のメモリを備えてもよい。

【0048】

HDD103は、OS(Operating System)やミドルウェアやアプリケーションソフトウェアなどのソフトウェアのプログラム、および、データを記憶する不揮発性の記憶装置である。プログラムには、セッション制御プログラムが含まれる。なお、Webサーバ100は、フラッシュメモリやSSD(Solid State Drive)などの他の種類の記憶装置を備えてもよく、複数の不揮発性の記憶装置を備えてもよい。

【0049】

画像信号処理部104は、CPU101からの命令に従って、Webサーバ100に接続されたディスプレイ111に画像を出力する。ディスプレイ111としては、CRT(Cathode Ray Tube)ディスプレイ、液晶ディスプレイ(LCD:Liquid Crystal Display)、プラズマディスプレイ、有機EL(OEL:Organic Electro-Luminescence)ディスプレイなどを用いることができる。ただし、ディスプレイ111が、Webサーバ100の筐体と一体に形成されていてもよい。

【0050】

入力信号処理部105は、Webサーバ100に接続された入力デバイス112から入力信号を取得し、CPU101に出力する。入力デバイス112としては、マウスやタッチパネルやタッチパッドやトラックボールなどのポインティングデバイス、キーボード、リモートコントローラ、ボタンスイッチなどを用いることができる。また、Webサーバ100に、複数の種類の入力デバイスが接続されていてもよい。ただし、入力デバイス112が、Webサーバ100の筐体と一体に形成されていてもよい。

【0051】

媒体リーダ106は、記録媒体113に記録されたプログラムやデータを読み取る読み取り装置である。記録媒体113として、例えば、フレキシブルディスク(FD:Flexib

10

20

30

40

50

le Disk)やHDDなどの磁気ディスク、CD(Compact Disc)やDVD(Digital Versatile Disc)などの光ディスク、光磁気ディスク(MO:Magneto-Optical disk)、半導体メモリなどを使用できる。媒体リーダ106は、例えば、記録媒体113から読み取ったプログラムやデータをRAM102またはHDD103に格納する。

【0052】

通信インタフェース107は、ネットワーク44に接続され、ネットワーク44を介して他の情報処理装置と通信を行うインタフェースである。通信インタフェース107は、スイッチなどの通信装置とケーブルで接続される有線通信インタフェースでもよいし、基地局と無線リンクで接続される無線通信インタフェースでもよい。

【0053】

図4は、第2の実施の形態のサーバアプリケーションの配備例を示す図である。

Webサーバ100は、リバースプロキシ120および実行環境150を実行する。Webサーバ100aは、リバースプロキシ120aおよび実行環境150aを実行する。Webサーバ100bは、リバースプロキシ120bおよび実行環境150bを実行する。サーバアプリケーション140が、最初はWebサーバ100aで実行されており、その後Webサーバ100に移動する場合を考える。なお、第2の実施の形態では、リバースプロキシ120、120bはサーバアプリケーション140と直接通信するため、Webサーバ100aはリバースプロキシ120aを実行しなくてもよい。

【0054】

Webサーバ100のIPアドレスは「10.x.x.40」である。Webサーバ100aのIPアドレスは「10.x.x.30」である。Webサーバ100bのIPアドレスは「10.x.x.50」である。サーバアプリケーション140がWebサーバ100aに配置されている場合、サーバアプリケーション140には「ws://10.x.x.30:8880」を含む実URLが付与される。サーバアプリケーション140がWebサーバ100に配置されている場合、サーバアプリケーション140には「ws://10.x.x.40:8880」を含む実URLが付与される。

【0055】

一方、リバースプロキシ120、120bには「ws://vurl:80」を含む仮想URLが付与される。実行環境150、150a、150bは、受信したアクセスをURLに応じて振り分ける。実行環境150は、「ws://vurl:80」を含むアクセスをリバースプロキシ120に転送する。サーバアプリケーション140がWebサーバ100に配置されている場合、実行環境150は、「ws://10.x.x.40:8880」を含むアクセスをサーバアプリケーション140に転送する。

【0056】

サーバアプリケーション140がWebサーバ100aに配置されている場合、実行環境150aは、「ws://10.x.x.30:8880」を含むアクセスをサーバアプリケーション140に転送する。実行環境150bは、「ws://vurl:80」を含むアクセスをリバースプロキシ120bに転送する。

【0057】

リバースプロキシ120、120bは、「ws://vurl:80」を含むアクセスを受信すると、その仮想URLをサーバアプリケーション140の実URLに変換してアクセスを転送する。サーバアプリケーション140がWebサーバ100aに配置されている場合、リバースプロキシ120、120aは、「ws://vurl:80」を「ws://10.x.x.30:8880」に変換してアクセスを転送する。一方、サーバアプリケーション140がWebサーバ100に配置されている場合、リバースプロキシ120、120aは、「ws://vurl:80」を「ws://10.x.x.40:8880」に変換してアクセスを転送する。これにより、仮想URLを指定したアクセスがサーバアプリケーション140に到達する。

【0058】

また、実行環境150、150a、150bは、配備計画サーバ200からの指示に応

10

20

30

40

50

じてサーバアプリケーション140を移動する。例えば、移動元のWebサーバの実行環境は、移動先のWebサーバの実行環境に対してサーバアプリケーション140のプログラムを送信する。また、第2の実施の形態では、サーバアプリケーション140の移動にあたって内部状態も引き継ぐようにする。このため、プログラムと併せて、サーバアプリケーション140の実行状況を示す一時データも送信する。移動元のWebサーバの実行環境は、サーバアプリケーション140を停止する。移動先のWebサーバの実行環境は、受信したプログラムとデータに基づいてサーバアプリケーション140を起動する。

【0059】

ところで、端末装置41, 42, 51, 52およびWebサーバ100, 100a, 100bは、ネットワーク層の通信プロトコルとしてIPを使用することができる。また、端末装置41, 42, 51, 52およびWebサーバ100, 100a, 100bは、トランスポート層の通信プロトコルとしてTCPを使用することができる。また、端末装置41, 42, 51, 52およびWebサーバ100, 100a, 100bは、更に上位の通信プロトコルとして、HTTPおよびWebソケットを使用することができる。端末装置41, 42, 51, 52のクライアントアプリケーションとサーバアプリケーション140とは、HTTPおよびWebソケットを用いて通信する。

【0060】

図5は、HTTPパケットとWebソケットパケットの例を示す図である。

(A) HTTPパケット61は、TCPコネクション上で送信される。HTTPパケット61は、IPヘッダ、TCPヘッダ、HTTPヘッダおよびHTTPペイロードを含む。IPヘッダは、宛先IPアドレスと送信元IPアドレスを含む。TCPヘッダは、宛先ポート番号と送信元ポート番号を含む。HTTPヘッダは、UpgradeおよびConnectionを含む。更に、HTTPリクエストのHTTPヘッダは、メソッドを含む。HTTPレスポンスのHTTPヘッダは、ステータス番号を含む。HTTPペイロードは、上り通信におけるPOSTデータや、下り通信におけるHTML (HyperText Markup Language) ファイルなどのデータを含む。

【0061】

メソッドは、GETやPOSTなどの命令とURLとを含む。例えば、GETメソッドは、URLが示すサーバアプリケーションに対してコンテンツを要求するときに用いられる。POSTメソッドは、URLが示すサーバアプリケーションに対してデータをアップロードするときに用いられる。ステータス番号は、サーバアプリケーションの処理結果を示す。例えば、「200」は、HTTPリクエストが正常に処理されたことを示す。「101」は、通信プロトコルをHTTPから他のものに切り替えることを示す。

【0062】

Upgradeの項目は、通信プロトコルを切り替える場合に切り替え後の通信プロトコルを示す。Connectionの項目は、そのHTTPメッセージの後にTCPコネクションをどのように取り扱うかを示す。例えば、「Close」は、TCPコネクションを切断することを示す。「Upgrade」は、TCPコネクションを維持し、そのTCPコネクション上で上位の通信プロトコルを変更することを示す。

【0063】

(B) Webソケットパケット62は、TCPコネクション上で送信される。Webソケットパケット62は、IPヘッダ、TCPヘッダ、Webソケットヘッダ (WSヘッダ) およびWebソケットペイロード (WSペイロード) を含む。IPヘッダとTCPヘッダは、HTTPパケット61と同じである。WSヘッダは、HTTPヘッダよりもサイズが小さく軽量なヘッダであり、OpCodeを含む。WSペイロードは、クライアントアプリケーションとサーバアプリケーションの一方から他方に送信されるデータを含む。

【0064】

OpCodeは、WSペイロードの形式や、送信元から宛先に対する命令を示す。OpCodeは、4ビットで表現することができる。例えば、「0 (0x0)」は、WSペイロードが前のパケットの続きであることを示す。「1 (0x1)」は、WSペイロードが

10

20

30

40

50

テキストデータを含むことを示す。「2 (0 x 2)」は、WSペイロードがバイナリデータを含むことを示す。「8 (0 x 8)」は、Webソケットによる通信の終了(クローズ)を示す。「9 (0 x 9)」は、疎通確認(ping)を示す。「10 (0 x A)」は、疎通応答(ping に対する応答)を示す。

【0065】

HTTPでは、ひとまとまりのHTTPメッセージの送信が完了する毎に、クライアントアプリケーションとサーバアプリケーションとの間のTCPコネクションが切断される。例えば、クライアントアプリケーションが一のWebページを要求するHTTPリクエストを送信し、サーバアプリケーションが要求されたWebページに関するデータを含むHTTPレスポンスを返信すると、両者の間のTCPコネクションは切断される。別のWebページを要求する場合には、改めてTCPコネクションを確立することになる。

10

【0066】

これに対し、Webソケットでは、クライアントアプリケーションがサーバアプリケーションの利用を開始してから終了するまでの間、原則として両者の間にTCPコネクションが維持される。サーバアプリケーションの利用を開始してから終了するまでの一連の通信をWebソケットセッション(WSセッション)とすることがある。Webソケットは、クライアントアプリケーションとサーバアプリケーションの間で小さなデータが頻繁に送信される場合や、サーバアプリケーションからクライアントアプリケーションに対してデータがプッシュされる場合に利用されることがある。サーバアプリケーションの一例として、チャットアプリケーションが挙げられる。

20

【0067】

クライアントアプリケーションおよびサーバアプリケーションそれぞれから見て、1つのWSセッションは1つのTCPコネクションと対応する。クライアントアプリケーションまたはサーバアプリケーションは、通信相手からOpCode = 0 x 8を含むWebソケットメッセージ(クローズ通知)を受信すると、WSセッションが終了または中断されたと認識する。すると、クライアントアプリケーションまたはサーバアプリケーションは、そのWSセッションに対応するTCPコネクションを切断する。

【0068】

クライアントアプリケーションがクローズ通知を受信した後、再びサーバアプリケーションのサービスを利用したい場合、改めてサーバアプリケーションとの間にTCPコネクションを確立することになる。クライアントアプリケーションは、TCPコネクション上でログインなどの手続を行うことが考えられる。このとき、前のWSセッションを再開できる(前の内部状態を引き継ぐことができる)場合もあるし、前のWSセッションは終了しており新たなWSセッションを開始することになる場合もある。

30

【0069】

図6は、Webソケットによる第1の通信例を示すシーケンス図である。

ここでは、サーバアプリケーション140がWebサーバ100aに配置されており、端末装置41がWebサーバ100aとWebソケットによって通信する場合を考える。また、端末装置41とWebサーバ100aとが、リバースプロキシ120を経由せずに直接通信すると仮定する。例えば、端末装置41のユーザがサーバアプリケーション140の実URLを入力した場合に、図6のような通信が発生する。

40

【0070】

端末装置41は、Webサーバ100aとの間でTCPの3ウェイハンドシェイク(TCPハンドシェイク)を実行することで、TCPコネクションを確立する(S110)。すなわち、端末装置41は、SYNパケット(SYN = 1, ACK = 0のパケット)をWebサーバ100aに送信する。Webサーバ100aは、SYN-ACKパケット(SYN = 1, ACK = 1のパケット)を端末装置41に送信する。端末装置41は、ACKパケット(SYN = 0, ACK = 1のパケット)をWebサーバ100aに送信する。SYNビットは接続要求を示し、ACKビットは接続要求に対する許可応答を示す。

【0071】

50

端末装置 4 1 は、TCP 接続上で WS ハンドシェイク要求を Web サーバ 1 0 0 a に送信する (S 1 1 1)。WS ハンドシェイク要求は、HTTP ヘッダの Upgrade を「websocket」に設定した HTTP リクエストである。Web サーバ 1 0 0 a は、WS ハンドシェイク要求を受信すると、TCP 接続上で WS ハンドシェイク応答を端末装置 4 1 に送信する (S 1 1 2)。WS ハンドシェイク応答は、HTTP ヘッダのステータス番号を「101」、Upgrade を「websocket」、Connection を「Upgrade」に設定した HTTP レスポンスである。

【0072】

なお、Web ソケットハンドシェイク (WS ハンドシェイク) は、TCP 接続上で Web ソケットの通信を可能にする (WS セッションを開始する) ための手続である。WS ハンドシェイクは、Opening ハンドシェイクと言うこともある。

10

【0073】

これにより、以降は端末装置 4 1 と Web サーバ 1 0 0 a の間で、アプリケーション層の通信プロトコルとして HTTP に代えて Web ソケットが使用される。例えば、端末装置 4 1 は、TCP 接続上で、Web ソケットのデータを Web サーバ 1 0 0 a に送信する (S 1 1 3)。Web サーバ 1 0 0 a は、端末装置 4 1 からのアクセスを待たずに端末装置 4 1 にデータを送信することもできる。

【0074】

Web ソケットの通信を終了する場合、Web サーバ 1 0 0 a は、TCP 接続上でクローズ通知を端末装置 4 1 に送信する (S 1 1 4)。クローズ通知は、前述の通り、WS ヘッダの OpCode を「0x8」に設定した Web ソケットメッセージである。端末装置 4 1 は、Web サーバ 1 0 0 a からクローズ通知を受信すると、端末装置 4 1 と Web サーバ 1 0 0 a との間の TCP 接続を切断する (S 1 1 5)。すなわち、端末装置 4 1 は、Web サーバ 1 0 0 a に FIN パケットを送信する。Web サーバ 1 0 0 a は、端末装置 4 1 に FIN パケットを返信する。

20

【0075】

TCP 接続は、ステップ S 1 1 0 で TCP ハンドシェイクが行われてから、ステップ S 1 1 5 で切断されるまで維持されることになる。TCP 接続上に形成される WS セッションは、ステップ S 1 1 2 で WS ハンドシェイクが完了してからステップ S 1 1 4 でクローズ通知が送信されるまで維持されることになる。

30

【0076】

次に、端末装置 4 1 がサーバアプリケーション 1 4 0 との間に TCP 接続を維持しているときに、サーバアプリケーション 1 4 0 を移動する場合の問題点を説明する。

図 7 は、Web ソケットによる第 2 の通信例を示すシーケンス図である。

【0077】

ここでは、サーバアプリケーション 1 4 0 は最初に Web サーバ 1 0 0 a に配置されており、その後 Web サーバ 1 0 0 に移動するものとする。また、図 6 と同様に、端末装置 4 1 と Web サーバ 1 0 0 , 1 0 0 a とが、リバースプロキシ 1 2 0 を経由せずに直接通信すると仮定する。

【0078】

端末装置 4 1 は、Web サーバ 1 0 0 a と TCP ハンドシェイクを行い、端末装置 4 1 と Web サーバ 1 0 0 a との間に TCP 接続を確立する (S 1 2 0)。端末装置 4 1 は、HTTP により WS ハンドシェイク要求を Web サーバ 1 0 0 a に送信する (S 1 2 1)。Web サーバ 1 0 0 a は、HTTP により WS ハンドシェイク応答を端末装置 4 1 に送信する (S 1 2 2)。端末装置 4 1 は、WS セッションの中で Web ソケットのデータを Web サーバ 1 0 0 a に送信する (S 1 2 3)。

40

【0079】

ここで、配備計画サーバ 2 0 0 がサーバアプリケーション 1 4 0 を Web サーバ 1 0 0 a から Web サーバ 1 0 0 に移動すると決定したとする。配備計画サーバ 2 0 0 が Web サーバ 1 0 0 a に移動を指示すると、上記の TCP 接続を維持できないため、W

50

e bサーバ100aはクローズ通知を端末装置41に送信する(S124)。端末装置41は、クローズ通知を受信するとTCPコネクションを切断する(S125)。

【0080】

クローズ通知を受信すると、端末装置41からは、サーバアプリケーション140側の都合によってサービスが突然利用できなくなった(例えば、サービスから強制的にログアウトさせられた)ように見える。端末装置41のクライアントアプリケーションは、例えば、サービスが利用不可になったことをユーザに通知する。サーバアプリケーション140のサービスを再び利用したい場合、端末装置41は、Webサーバ100にアクセスすることになる。このとき、端末装置41のユーザが、適切なURLを入力する、ユーザIDとパスワードを指定してログインするなどの操作を要する場合がある。

10

【0081】

端末装置41は、Webサーバ100とTCPハンドシェイクを行い、端末装置41とWebサーバ100との間にTCPコネクションを確立する(S126)。端末装置41は、HTTPによりWSハンドシェイク要求をWebサーバ100に送信する(S127)。Webサーバ100は、HTTPによりWSハンドシェイク応答を端末装置41に送信する(S128)。端末装置41は、WSセッションの中でWebソケットのデータをWebサーバ100に送信する(S129)。

【0082】

正常にWSセッションを終了する場合、Webサーバ100は、クローズ通知を端末装置41に送信する(S130)。端末装置41は、クローズ通知を受信するとTCPコネクションを切断する(S131)。ただし、端末装置41からWebサーバ100にクローズ通知を送信することも可能である。

20

【0083】

ステップS120からステップS125まで1つのTCPコネクションが維持され、ステップS126からステップS131まで1つのTCPコネクションが維持されている。一方、ステップS125とステップS126の間はTCPコネクションが存在しない。

【0084】

また、端末装置41から見て、ステップS122からステップS124までとステップS128からステップS130までは、WSセッションが維持されている。一方、端末装置41から見て、ステップS125とステップS126の間はWSセッションが利用不可である。なお、端末装置41からは、ステップS124でWSセッションが終了し、ステップS128で新たなWSセッションが開始されたように見えることがある。また、端末装置41からは、ステップS124でWSセッションが中断され、ステップS128で当該WSセッションが再開されたように見えることがある。

30

【0085】

このように、Webソケット通信を行うサーバアプリケーション140を移動すると、端末装置41のユーザの利便性が低下するおそれがある。そこで、第2の実施の形態では、リバースプロキシ120を利用することで、端末装置41からTCPコネクションの切断を隠蔽し、移動中もWSセッションが継続しているように見せるようにする。

【0086】

図8は、Webソケットによる第3の通信例を示すシーケンス図である。

40

ここでは、図7と同様に、サーバアプリケーション140が最初にWebサーバ100aに配置されており、その後Webサーバ100に移動するものとする。ただし、端末装置41は、リバースプロキシ120経由でサーバアプリケーション140と通信する。

【0087】

端末装置41は、リバースプロキシ120とTCPハンドシェイクを行い、端末装置41とリバースプロキシ120との間にTCPコネクションを確立する(S140)。端末装置41は、サーバアプリケーション140の仮想URLを指定したWSハンドシェイク要求をリバースプロキシ120に送信する(S141)。リバースプロキシ120は、仮想URLをサーバアプリケーション140の実URLに変換する。

50

【 0 0 8 8 】

リバースプロキシ 1 2 0 は、Webサーバ 1 0 0 a に配置されたサーバアプリケーション 1 4 0 と TCP ハンドシェイクを行い、リバースプロキシ 1 2 0 とサーバアプリケーション 1 4 0 との間に TCP コネクションを確立する (S 1 4 2)。リバースプロキシ 1 2 0 は、サーバアプリケーション 1 4 0 の実 URL を指定した WS ハンドシェイク要求をサーバアプリケーション 1 4 0 に送信する (S 1 4 3)。

【 0 0 8 9 】

サーバアプリケーション 1 4 0 は、WS ハンドシェイク応答をリバースプロキシ 1 2 0 に送信する。リバースプロキシ 1 2 0 は、WS ハンドシェイク応答を端末装置 4 1 に転送する (S 1 4 4)。端末装置 4 1 は、Web ソケットのデータをリバースプロキシ 1 2 0 に送信する。リバースプロキシ 1 2 0 は、Web ソケットのデータをサーバアプリケーション 1 4 0 に転送する (S 1 4 5)。

10

【 0 0 9 0 】

ここで、配備計画サーバ 2 0 0 がサーバアプリケーション 1 4 0 を Webサーバ 1 0 0 a から Webサーバ 1 0 0 に移動すると決定したとする。配備計画サーバ 2 0 0 がサーバアプリケーション 1 4 0 に移動指示を送信すると、サーバアプリケーション 1 4 0 はクローズ通知をリバースプロキシ 1 2 0 に送信する (S 1 4 6)。リバースプロキシ 1 2 0 は、受信したクローズ通知を終端する。すなわち、リバースプロキシ 1 2 0 は、クローズ通知を端末装置 4 1 に転送せずに破棄する。なお、後述するように、リバースプロキシ 1 2 0 は、配備計画サーバ 2 0 0 から移動開始通知を受信した場合のみクローズ通知を終端し、それ以外の場合は通常の Web ソケットメッセージと同様に転送することになる。

20

【 0 0 9 1 】

リバースプロキシ 1 2 0 は、クローズ通知を受信すると、リバースプロキシ 1 2 0 とサーバアプリケーション 1 4 0 との間の TCP コネクションを切断する (S 1 4 7)。一方、リバースプロキシ 1 2 0 は、端末装置 4 1 とリバースプロキシ 1 2 0 との間の TCP コネクションは切断しない。クローズ通知が端末装置 4 1 に転送されないため、端末装置 4 1 によっても当該 TCP コネクションは切断されない。

【 0 0 9 2 】

端末装置 4 1 は、WS セッションが継続していると認識しているため、Web ソケットのデータをリバースプロキシ 1 2 0 に送信する (S 1 4 8)。リバースプロキシ 1 2 0 は、サーバアプリケーション 1 4 0 との間に TCP コネクションが存在しないため、受信したデータをバッファに保存しておく。

30

【 0 0 9 3 】

そして、リバースプロキシ 1 2 0 は、後述するように配備計画サーバ 2 0 0 から移動完了通知を受信すると、Webサーバ 1 0 0 に移動したサーバアプリケーション 1 4 0 と TCP ハンドシェイクを行い、TCP コネクションを確立する (S 1 4 9)。この TCP コネクションは、Webサーバ 1 0 0 の内部に形成されたコネクションである。リバースプロキシ 1 2 0 は、WS ハンドシェイク要求をサーバアプリケーション 1 4 0 に送信する (S 1 5 0)。サーバアプリケーション 1 4 0 は、WS ハンドシェイク応答をリバースプロキシ 1 2 0 に送信する (S 1 5 1)。リバースプロキシ 1 2 0 は、バッファに保存されたデータをサーバアプリケーション 1 4 0 に転送する (S 1 5 2)。

40

【 0 0 9 4 】

なお、前述のように、サーバアプリケーション 1 4 0 を Webサーバ 1 0 0 a から Webサーバ 1 0 0 に移動するとき、WS セッションの状態を示す一時データ (セッションデータ) も移動する。このため、サーバアプリケーション 1 4 0 は、ステップ S 1 4 6 のクローズ通知を送信する前の文脈 (コンテキスト) を引き継ぐことができる。

【 0 0 9 5 】

正常に WS セッションを終了する場合、サーバアプリケーション 1 4 0 は、クローズ通知をリバースプロキシ 1 2 0 に送信する。リバースプロキシ 1 2 0 は、配備計画サーバ 2 0 0 から移動開始通知を受信していないため、通常の Web ソケットメッセージと同様に

50

クローズ通知を端末装置 4 1 に転送する (S 1 5 3) 。また、リバースプロキシ 1 2 0 は、リバースプロキシ 1 2 0 とサーバアプリケーション 1 4 0 との間の T C P コネクションを切断する。端末装置 4 1 は、クローズ通知を受信すると、端末装置 4 1 とリバースプロキシ 1 2 0 との間の T C P コネクションを切断する (S 1 5 4) 。

【 0 0 9 6 】

サーバアプリケーション 1 4 0 から見て、ステップ S 1 4 2 からステップ S 1 4 7 まで 1 つの T C P コネクションが維持され、ステップ S 1 4 9 からステップ S 1 5 4 まで 1 つの T C P コネクションが維持されている。サーバアプリケーション 1 4 0 から見て、ステップ S 1 4 7 とステップ S 1 4 9 の間は T C P コネクションが存在しない。

【 0 0 9 7 】

これに対し、端末装置 4 1 から見て、ステップ S 1 4 0 からステップ S 1 5 4 まで T C P コネクションが切断されずに維持されている。また、端末装置 4 1 からは、ステップ S 1 4 4 からステップ S 1 5 3 まで W S セッションが継続しているように見える。また、サーバアプリケーション 1 4 0 の移動の際にはセッションデータも移動する。よって、端末装置 4 1 は、サーバアプリケーション 1 4 0 の移動を意識せずに通信を継続できる。

【 0 0 9 8 】

図 9 は、第 2 の実施の形態の移動制御の通信例を示すシーケンス図である。

配備計画サーバ 2 0 0 は、サーバアプリケーション 1 4 0 を W e b サーバ 1 0 0 a から W e b サーバ 1 0 0 に移動することを決定する。すると、配備計画サーバ 2 0 0 は、W e b サーバ 1 0 0 のリバースプロキシ 1 2 0 と W e b サーバ 1 0 0 b のリバースプロキシ 1 2 0 b に、移動開始通知を送信する (S 1 6 0) 。移動開始通知は、移動するサーバアプリケーション 1 4 0 を特定できるように仮想 U R L を含む。リバースプロキシ 1 2 0 , 1 2 0 b は、移動開始通知に対する応答を配備計画サーバ 2 0 0 に送信する (S 1 6 1) 。

【 0 0 9 9 】

配備計画サーバ 2 0 0 は、サーバアプリケーション 1 4 0 に停止指示を送信する。サーバアプリケーション 1 4 0 は、W e b ソケットのために維持している T C P コネクション毎にクローズ通知を送信する (S 1 6 2) 。例えば、サーバアプリケーション 1 4 0 は、リバースプロキシ 1 2 0 , 1 2 0 b に 1 または 2 以上のクローズ通知を送信する。このクローズ通知は、図 8 のステップ S 1 4 6 のクローズ通知に対応する。

【 0 1 0 0 】

配備計画サーバ 2 0 0 は、W e b サーバ 1 0 0 a の実行環境 1 5 0 a に移動指示を送信する。実行環境 1 5 0 a は、サーバアプリケーション 1 4 0 のプログラムとその内部状態を示すセッションデータを W e b サーバ 1 0 0 に送信する (S 1 6 3) 。実行環境 1 5 0 a は、サーバアプリケーション 1 4 0 を停止させる。W e b サーバ 1 0 0 の実行環境 1 5 0 は、プログラムとセッションデータによりサーバアプリケーション 1 4 0 を起動する。

【 0 1 0 1 】

配備計画サーバ 2 0 0 は、リバースプロキシ 1 2 0 , 1 2 0 b に移動完了通知を送信する (S 1 6 4) 。移動完了通知は、移動したサーバアプリケーション 1 4 0 を特定できるように仮想 U R L を含む。また、移動完了通知は、メッセージをサーバアプリケーション 1 4 0 に転送できるように移動後の実 U R L (移動前のものとは異なる) を含む。

【 0 1 0 2 】

図 1 0 は、配備計画サーバの状態遷移例を示す図である。

配備計画サーバ 2 0 0 は、最初は起動されていない停止状態 7 1 にある。配備計画サーバ 2 0 0 は、起動されると停止状態 7 1 から定常状態 7 2 に遷移する。定常状態 7 2 は、サーバアプリケーションの現在の配置を維持している状態である。配備計画サーバ 2 0 0 は、定常状態 7 2 において、W e b サーバ 1 0 0 , 1 0 0 a , 1 0 0 b の負荷やサーバアプリケーションの通信の遅延時間などを示す負荷情報を収集する。

【 0 1 0 3 】

配備計画サーバ 2 0 0 は、負荷情報に基づいてサーバアプリケーションの移動を要すると判断すると、定常状態 7 2 から移動計画中状態 7 3 に遷移する。配備計画サーバ 2 0 0

10

20

30

40

50

は、移動計画中状態 7 3 において、移動するサーバアプリケーションと移動先の Web サーバを示す移動計画を決定する。移動計画が決定すると、配備計画サーバ 2 0 0 は、移動計画中状態 7 3 から移動開始通知状態 7 4 に遷移する。

【 0 1 0 4 】

配備計画サーバ 2 0 0 は、移動開始通知状態 7 4 において、各リバースプロキシに移動開始通知を送信し、移動開始通知に対する応答を待つ。全てのリバースプロキシから移動開始通知に対する応答を受信すると、配備計画サーバ 2 0 0 は、移動開始通知状態 7 4 から移動中状態 7 5 に遷移する。配備計画サーバ 2 0 0 は、移動中状態 7 5 において、移動するサーバアプリケーションに対して停止指示を送信する。また、配備計画サーバ 2 0 0 は、移動元の Web サーバに対して移動指示を送信する。

10

【 0 1 0 5 】

サーバアプリケーションの移動が完了すると、配備計画サーバ 2 0 0 は、移動中状態 7 5 から移動完了通知状態 7 6 に遷移する。配備計画サーバ 2 0 0 は、移動完了通知状態 7 6 において、各リバースプロキシに移動完了通知を送信する。そして、配備計画サーバ 2 0 0 は、移動完了通知状態 7 6 から定常状態 7 2 に遷移する。このようにして、配備計画サーバ 2 0 0 は、サーバアプリケーションの配置を継続的に見直し、Web サーバ 1 0 0 , 1 0 0 a , 1 0 0 b へのサーバアプリケーションの配置を最適化する。

【 0 1 0 6 】

次に、リバースプロキシ 1 2 0 , 1 2 0 b の機能について説明する。以下では、リバースプロキシ 1 2 0 , 1 2 0 b のうち代表してリバースプロキシ 1 2 0 を説明する。リバースプロキシ 1 2 0 b も、リバースプロキシ 1 2 0 と同様の機能を有する。

20

【 0 1 0 7 】

図 1 1 は、第 2 の実施の形態のリバースプロキシの機能例を示すブロック図である。

リバースプロキシ 1 2 0 は、制御情報記憶部 1 2 1、データ記憶部 1 2 2 および状態制御部 1 2 3 を有する。また、リバースプロキシ 1 2 0 は、サーバ機能部 1 3 1、ソケット検索部 1 3 2 , 1 3 8、端末情報保存部 1 3 3、データ保留部 1 3 4、宛先解決部 1 3 5、URL 書換部 1 3 6、クライアント機能部 1 3 7 およびクローズ無効化部 1 3 9 を有する。制御情報記憶部 1 2 1 およびデータ記憶部 1 2 2 は、例えば、RAM 1 0 2 または HDD 1 0 3 に確保した記憶領域を用いて実装される。他のユニットは、例えば、CPU 1 0 1 が実行するプログラムモジュールを用いて実装される。

30

【 0 1 0 8 】

制御情報記憶部 1 2 1 は、HTTP メッセージや Web ソケットメッセージの転送に用いられる各種の制御情報を記憶する。制御情報の詳細は後述する。データ記憶部 1 2 2 は、サーバアプリケーション 1 4 0 が移動中である場合に、端末装置 4 1 , 4 2 から受信したメッセージを一時的に格納するバッファである。状態制御部 1 2 3 は、配備計画サーバ 2 0 0 から、移動開始通知や移動完了通知などの各種の通知を受信する。状態制御部 1 2 3 は、受信した通知に応じて、制御情報記憶部 1 2 1 に記憶された制御情報を更新する。

【 0 1 0 9 】

サーバ機能部 1 3 1 は、端末装置 4 1 , 4 2 に対してサーバアプリケーションとして振る舞う。サーバ機能部 1 3 1 は、端末装置 4 1 , 4 2 からの要求に応じて端末装置 4 1 , 4 2 との間に TCP コネクションを確立する。サーバ機能部 1 3 1 は、TCP コネクション毎に、論理的な通信インタフェースであるソケットを生成する。サーバ機能部 1 3 1 は、ソケットの番号によって TCP コネクションを区別する。サーバ機能部 1 3 1 は、ソケットから（端末装置 4 1 , 4 2 から）メッセージを受信する。また、サーバ機能部 1 3 1 は、ソケットを指定して（端末装置 4 1 , 4 2 に対して）メッセージを送信する。

40

【 0 1 1 0 】

ソケット検索部 1 3 2 は、サーバ機能部 1 3 1 がメッセージを受信すると、制御情報記憶部 1 2 1 に記憶された制御情報を参照して、メッセージが受信されたソケットに対応するクライアント機能部 1 3 7 側のソケットを検索する。検索されるソケットは、リバースプロキシ 1 2 0 とサーバアプリケーション 1 4 0 との間の TCP コネクションに対応する

50

ソケットである。端末情報保存部 133 は、受信されたメッセージが HTTP メッセージである場合、HTTP ヘッダを制御情報記憶部 121 に保存する。

【0111】

データ保留部 134 は、制御情報記憶部 121 に記憶された制御情報を参照して、サーバアプリケーション 140 が移動中であるか判定する。サーバアプリケーション 140 が移動中である場合、データ保留部 134 は、受信されたメッセージをデータ記憶部 122 に保存する。サーバアプリケーション 140 の移動が完了すると、データ保留部 134 は、データ記憶部 122 に記憶されているメッセージを、サーバアプリケーション 140 に転送すべきメッセージとして取り出す。ただし、記憶されたメッセージが Web ソケットメッセージである場合、データ保留部 134 は、それに先立ってサーバアプリケーション 140 に送信すべき HTTP メッセージとして、WS ハンドシェイク要求を生成する。WS ハンドシェイク要求には、リバースプロキシ 120 が端末装置 41, 42 であるかのように振る舞うため、端末情報保存部 133 が保存した HTTP ヘッダが付加される。

10

【0112】

宛先解決部 135 は、転送すべきメッセージが HTTP メッセージである場合、制御情報記憶部 121 に記憶された制御情報を参照して、HTTP メッセージに含まれる仮想 URL に対応する実 URL を検索する。また、宛先解決部 135 は、転送すべきメッセージが Web ソケットメッセージである場合、制御情報記憶部 121 に記憶された制御情報を参照して、サーバアプリケーション 140 との通信に用いられるクライアント機能部 137 側のソケットを検索する。URL 書換部 136 は、HTTP メッセージに含まれる仮想 URL を、宛先解決部 135 が検索した実 URL に書き換える。

20

【0113】

クライアント機能部 137 は、サーバアプリケーション 140 に対してクライアントアプリケーションとして振る舞う。クライアント機能部 137 は、サーバアプリケーション 140 との間に TCP コネクションを確立する。クライアント機能部 137 は、クライアント機能部 137 は、TCP コネクション毎に、論理的な通信インタフェースであるソケットを生成する。クライアント機能部 137 は、ソケットを指定して（サーバアプリケーション 140 に対して）メッセージを送信する。また、クライアント機能部 137 は、ソケットから（サーバアプリケーション 140 から）メッセージを受信する。

【0114】

30

なお、リバースプロキシ 120 は、サーバ機能部 131 側が使用するソケットの番号の範囲と、クライアント機能部 137 が使用するソケットの番号の範囲とを予め決めておく。リバースプロキシ 120 は、メッセージが受信されたソケットの番号から、そのメッセージがサーバ機能部 131 によって処理されるべきものか、クライアント機能部 137 によって処理されるべきものか区別することができる。

【0115】

ソケット検索部 138 は、クライアント機能部 137 がメッセージを受信すると、制御情報記憶部 121 に記憶された制御情報を参照して、メッセージが受信されたソケットに対応するサーバ機能部 131 側のソケットを検索する。

【0116】

40

クローズ無効化部 139 は、制御情報記憶部 121 に記憶された制御情報を参照して、受信されたメッセージが Web ソケットメッセージであり、かつ、サーバアプリケーション 140 が移動中であるか判定する。すなわち、クローズ無効化部 139 は、サーバアプリケーション 140 の移動時のクローズ通知を受信したか判定する。上記条件に該当する場合、クローズ無効化部 139 は、受信されたクローズ通知を終端する。すなわち、クローズ無効化部 139 は、クローズ通知をサーバ機能部 131 に出力せずに破棄する。なお、クローズ通知が受信された場合、クライアント機能部 137 は、サーバアプリケーション 140 との間の TCP コネクションを切断してソケットを削除する。

【0117】

配備計画サーバ 200 は、制御情報記憶部 221、移動制御部 231、移動開始通知部

50

232 および移動完了通知部233を有する。制御情報記憶部221は、例えば、RAMまたはHDDに確保した記憶領域を用いて実装される。移動制御部231、移動開始通知部232および移動完了通知部233は、例えば、CPUが実行するプログラムモジュールを用いて実装される。

【0118】

制御情報記憶部221は、サーバアプリケーション140の移動の制御に用いられる各種の制御情報を記憶する。制御情報の詳細は後述する。移動制御部231は、Webサーバ100、100a、100bの負荷やサーバアプリケーション140の通信遅延を監視する。移動制御部231は、負荷や通信遅延が所定条件を満たすと、サーバアプリケーション140を移動すること、および、移動先のWebサーバを決定する。

10

【0119】

移動開始通知部232は、サーバアプリケーション140の移動を開始する前に、リバースプロキシ120、120bに対して移動開始通知を送信する。移動完了通知部233は、サーバアプリケーション140の移動が完了すると、リバースプロキシ120、120bに対して移動完了通知を送信する。

【0120】

図12は、リバースプロキシが有する制御テーブルの例を示す図である。

端末情報テーブル124、ソケット対応テーブル125、移動管理テーブル126およびURL対応テーブル127が、制御情報記憶部121に記憶されている。

【0121】

端末情報テーブル124は、端末側ソケットとHTTPヘッダの項目を有する。端末側ソケットの項目には、サーバ機能部131側のソケットの番号が登録される。HTTPヘッダの項目には、端末側ソケットの項目が示すソケットから受信されたHTTPメッセージに含まれるHTTPヘッダが登録される。1つのソケットから複数のHTTPメッセージが受信された場合には、HTTPヘッダの項目には、最新のHTTPメッセージに含まれるHTTPヘッダのみ登録されていけばよい。

20

【0122】

ソケット対応テーブル125は、端末側ソケット、Upgrade状態、サーバ側ソケットおよびポインタの項目を有する。端末側ソケットの項目には、サーバ機能部131側のソケットの番号が登録される。Upgrade状態の項目には、端末側ソケットの項目が示すソケット上の通信プロトコルが、HTTPからWebソケットにアップグレードされたか否かを示すフラグが登録される。Upgrade状態の初期値はFalseである。Upgrade状態がFalseであることは、通信プロトコルがHTTPであることを示す。Upgrade状態がTrueであることは、通信プロトコルがWebソケットであることを示す。

30

【0123】

サーバ側ソケットの項目には、クライアント機能部137側のソケットの番号が登録される。サーバ機能部131側の1つのソケットに対して、クライアント機能部137側の1つのソケットが対応付けられる。すなわち、リバースプロキシ120の端末装置41、42側の1つのTCPコネクションに対して、リバースプロキシ120のサーバアプリケーション140側の1つのTCPコネクションが対応付けられる。ポインタの項目には、移動管理テーブル126の1つのレコードを指す識別情報が登録される。識別情報として、後述する移動管理テーブル126のインデックスが用いられる。

40

【0124】

移動管理テーブル126は、インデックス、仮想URLおよび移動状態の項目を有する。インデックスは、移動管理テーブル126のレコードを識別する番号である。仮想URLの項目には、サーバアプリケーション140に付与された仮想URLが登録される。仮想URLは、例えば、配備計画サーバ200により決定される。移動状態の項目には、サーバアプリケーション140が移動中か否かを示すフラグが登録される。移動状態の初期値はFalseである。移動状態がFalseであることは、サーバアプリケーション1

50

40が移動中でないことを示す。移動状態がTrueであることは、サーバアプリケーション140が移動中であることを示す。

【0125】

URL対応テーブル127は、仮想URLおよび実URLの項目を有する。仮想URLの項目には、サーバアプリケーション140に付与された仮想URLが登録される。実URLの項目には、サーバアプリケーション140に付与された実URLが登録される。実URLは、サーバアプリケーション140が配置されるWebサーバに依存する。

【0126】

図13は、配備計画サーバが有する制御テーブルの例を示す図である。

プロトコルテーブル222、配備管理テーブル223およびプロキシ管理テーブル224が、制御情報記憶部221に記憶されている。

10

【0127】

プロトコルテーブル222は、アプリケーションIDおよびプロトコルの項目を有する。アプリケーションIDの項目には、サーバアプリケーション140を識別する識別情報が登録される。アプリケーションIDは、仮想URLまたは実URLに含まれるパス名であってもよい。プロトコルの項目には、サーバアプリケーション140が使用し得るアプリケーション層の通信プロトコルの名称が登録される。サーバアプリケーション140がWebソケットを使用し得る場合、プロトコルの項目にWebソケットが登録される。

【0128】

配備管理テーブル223は、アプリケーションID、仮想URLおよびサーバIDの項目を有する。アプリケーションIDの項目には、サーバアプリケーション140を識別する識別情報が登録される。仮想URLの項目には、サーバアプリケーション140に付与された仮想URLが登録される。サーバIDの項目には、サーバアプリケーション140が配置されたWebサーバを識別する識別情報が登録される。サーバIDは、Webサーバのホスト名またはIPアドレスであってもよい。

20

【0129】

プロキシ管理テーブル224は、サーバIDおよびプロキシURLの項目を有する。サーバIDの項目には、リバースプロキシ120, 120a, 120bそれぞれについて、当該リバースプロキシが配置されたWebサーバを識別する識別情報が登録される。サーバIDは、Webサーバのホスト名またはIPアドレスであってもよい。プロキシURLの項目には、リバースプロキシ120, 120a, 120bそれぞれについて、当該リバースプロキシに付与された実URLが登録される。なお、第2の実施の形態では、リバースプロキシ120, 120bはサーバアプリケーション140と直接通信するため、各リバースプロキシの実URLはリバースプロキシ120, 120bに通知されない。

30

【0130】

次に、リバースプロキシ120の処理手順について説明する。リバースプロキシ120bも、リバースプロキシ120と同様の処理を実行し得る。

図14は、上り通信転送の手順例を示すフローチャートである。

【0131】

(S210)サーバ機能部131は、メッセージを受信する。ソケット検索部132は、ソケット対応テーブル125から、メッセージが受信されたソケットの番号(受信ソケット番号)を端末側ソケットとして含むレコードを検索する。

40

【0132】

(S211)ソケット検索部132は、ステップS210において該当するレコードが検索されたか判断する。該当するレコードが検索された場合はステップS212に処理が進み、検索されなかった場合はステップS213に処理が進む。

【0133】

(S212)ソケット検索部132は、検索されたレコードに含まれるUpgrade状態がTrueであるか判断する。Upgrade状態がTrueである場合はステップS219に処理が進み、Falseである場合はステップS213に処理が進む。

50

【 0 1 3 4 】

(S 2 1 3) 端末情報保存部 1 3 3 は、受信されたメッセージが H T T P メッセージであると判断する。端末情報保存部 1 3 3 は、受信された H T T P メッセージに含まれる H T T P ヘッダを、受信ソケット番号と対応付けて端末情報テーブル 1 2 4 に登録する。

【 0 1 3 5 】

(S 2 1 4) データ保留部 1 3 4 は、受信された H T T P メッセージによって指定された U R L (仮想 U R L) を抽出する。データ保留部 1 3 4 は、移動管理テーブル 1 2 6 から、抽出した仮想 U R L を含むレコードを検索する。

【 0 1 3 6 】

(S 2 1 5) データ保留部 1 3 4 は、ステップ S 2 1 4 で検索されたレコードに含まれる移動状態が T r u e であるか判断する。移動状態が T r u e である場合はステップ S 2 1 6 に処理が進み、 F a l s e である場合はステップ S 2 1 7 に処理が進む。

10

【 0 1 3 7 】

(S 2 1 6) データ保留部 1 3 4 は、受信された H T T P メッセージをデータ記憶部 1 2 2 に格納して転送を保留する。データ保留部 1 3 4 は、ステップ S 2 1 4 で検索されたレコードの移動状態を監視し、移動状態が T r u e から F a l s e に変わるのを待つ。

【 0 1 3 8 】

(S 2 1 7) 宛先解決部 1 3 5 は、 H T T P メッセージによって指定された仮想 U R L を抽出する。宛先解決部 1 3 5 は、 U R L 対応テーブル 1 2 7 から、抽出した仮想 U R L に対応する実 U R L を検索する。 U R L 書換部 1 3 6 は、 H T T P メッセージに含まれる仮想 U R L を、検索された実 U R L に置換する。

20

【 0 1 3 9 】

(S 2 1 8) クライアント機能部 1 3 7 は、ソケット対応テーブル 1 2 5 から、受信ソケット番号を端末側ソケットとして含むレコードを検索する。クライアント機能部 1 3 7 は、検索されたレコードに、サーバ側ソケットが登録済みであるか判断する。サーバ側ソケットが登録済みである場合はステップ S 2 2 5 に処理が進み、サーバ側ソケットがまだ登録されていない場合はステップ S 2 2 3 に処理が進む。

【 0 1 4 0 】

(S 2 1 9) データ保留部 1 3 4 は、受信されたメッセージが W e b ソケットメッセージであると判断する。データ保留部 1 3 4 は、ステップ S 2 1 0 で検索されたレコードに含まれるポインタを辿り、ポインタが指すレコードを移動管理テーブル 1 2 6 から取得する。データ保留部 1 3 4 は、取得したレコードに含まれる移動状態を確認する。

30

【 0 1 4 1 】

(S 2 2 0) データ保留部 1 3 4 は、ステップ S 2 1 9 で確認した移動状態が T r u e であるか判断する。移動状態が T r u e である場合はステップ S 2 2 2 に処理が進み、 F a l s e である場合はステップ S 2 2 1 に処理が進む。

【 0 1 4 2 】

(S 2 2 1) 宛先解決部 1 3 5 は、ソケット対応テーブル 1 2 5 から、受信ソケット番号を端末側ソケットとして含むレコードを検索する。宛先解決部 1 3 5 は、検索されたレコードにサーバ側ソケットとして含まれるソケット番号 (サーバ側ソケット番号) を取得する。そして、ステップ S 2 2 5 に処理が進む。

40

【 0 1 4 3 】

(S 2 2 2) データ保留部 1 3 4 は、受信された W e b ソケットメッセージをデータ記憶部 1 2 2 に格納して転送を保留する。データ保留部 1 3 4 は、ステップ S 2 1 9 の移動状態を監視し、移動状態が T r u e から F a l s e に変わるのを待つ。

【 0 1 4 4 】

(S 2 2 3) クライアント機能部 1 3 7 は、メッセージの転送先のサーバアプリケーションとの間に T C P コネクションを確立し、サーバ側ソケットを生成する。 T C P コネクションは、 T C P の 3 ウェイハンドシェイクによって確立される。すなわち、クライアント機能部 1 3 7 は、 S Y N パケットを送信し、 S Y N - A C K パケットを受信し、 A C K

50

パケットを送信する。SYNパケットの宛先IPアドレスには、実URLに含まれるホスト名からDNSによって変換されるIPアドレス、または、実URLに含まれるIPアドレスが用いられる。SYNパケットの宛先ポート番号には、実URLに含まれるポート番号（省略されている場合は80番）が用いられる。

【0145】

(S224)クライアント機能部137は、ステップS223で生成したサーバ側ソケットの番号を、受信ソケット番号と対応付けてソケット対応テーブル125に登録する。また、クライアント機能部137は、登録したサーバ側ソケット番号に対応するポイントが空であるか（このレコードが今回新規に追加されたものか）判断する。ポイントが空である場合、クライアント機能部137は、ステップS214で検索された移動管理テーブル126のレコードに含まれるインデックスを、登録したサーバ側ソケット番号に対応するポイントとしてソケット対応テーブル125に登録する。

10

【0146】

(S225)クライアント機能部137は、ステップS218で検索されたソケット、ステップS223で生成したソケット、または、ステップS221で検索されたソケットを用いて、メッセージを送信する。なお、後述するように、送信するメッセージには、サーバ機能部131が受信したメッセージの他、データ保留部134が生成したWSハンドシェイク要求が含まれることがある。

【0147】

図15は、下り通信転送の手順例を示すフローチャートである。

20

(S230)クライアント機能部137は、メッセージを受信する。ソケット検索部138は、ソケット対応テーブル125から、メッセージが受信されたソケットの番号（受信ソケット番号）をサーバ側ソケットとして含むレコードを検索する。

【0148】

(S231)クローズ無効化部139は、検索されたレコードに含まれるUpgrade状態がTrueであるか判断する。Upgrade状態がTrueである場合はステップS234に処理が進み、Falseである場合はステップS232に処理が進む。

【0149】

(S232)サーバ機能部131は、受信されたメッセージがHTTPメッセージであると判断する。サーバ機能部131は、受信されたHTTPメッセージに含まれるHTTPヘッダからUpgradeの値を抽出し、Upgradeが「websocket」であるか判断する。Upgradeが「websocket」である場合はステップS233に処理が進み、それ以外の場合はステップS236に処理が進む。

30

【0150】

(S233)サーバ機能部131は、ステップS230で検索されたレコードのUpgrade状態をTrueに更新する。そして、ステップS236に処理が進む。

(S234)クローズ無効化部139は、受信されたメッセージがWebソケットメッセージであると判断する。クローズ無効化部139は、ステップS230で検索されたレコードのポイントを辿り、ポイントが指すレコードを移動管理テーブル126から取得する。クローズ無効化部139は、取得したレコードに含まれる移動状態を確認する。

40

【0151】

(S235)クローズ無効化部139は、ステップS234で確認した移動状態がTrueであるか判断する。移動状態がTrueである場合はステップS237に処理が進み、Falseである場合はステップS236に処理が進む。

【0152】

(S236)サーバ機能部131は、ステップS230で検索されたレコードが示す端末側ソケットを用いてメッセージを転送する。そして、下り通信転送が終了する。

(S237)クローズ無効化部139は、受信されたWebソケットメッセージ（クローズ通知）を終端する。すなわち、クローズ無効化部139は、クローズ通知をサーバ機能部131に出力せずに破棄する。

50

【 0 1 5 3 】

(S 2 3 8) サーバ機能部 1 3 1 は、ステップ S 2 3 0 で検索されたレコードのサーバ側ソケット番号をソケット対応テーブル 1 2 5 から削除する。

(S 2 3 9) クライアント機能部 1 3 7 は、Webソケットメッセージを受信したソケットに対応するTCPコネクションを切断する。すなわち、クライアント機能部 1 3 7 は、当該ソケットを指定してFINパケットを送信し、応答としてFINパケットを受信する。クライアント機能部 1 3 7 は、当該ソケットを削除する。

【 0 1 5 4 】

図 1 6 は、プロキシ状態制御の手順例を示すフローチャートである。

(S 2 4 0) 状態制御部 1 2 3 は、配備計画サーバ 2 0 0 から通知を受信する。

(S 2 4 1) 状態制御部 1 2 3 は、ステップ S 2 4 0 で受信した通知が移動開始通知であるか判断する。受信した通知が移動開始通知である場合はステップ S 2 4 2 に処理が進み、それ以外の場合はステップ S 2 4 6 に処理が進む。

【 0 1 5 5 】

(S 2 4 2) 状態制御部 1 2 3 は、受信した移動開始通知に含まれている仮想URL (指定された仮想URL) を抽出する。状態制御部 1 2 3 は、移動管理テーブル 1 2 6 から、抽出した仮想URLを含むレコードを検索する。

【 0 1 5 6 】

(S 2 4 3) 状態制御部 1 2 3 は、ステップ S 2 4 2 で検索されたレコードに含まれる移動状態をTrueに更新する。

(S 2 4 4) 状態制御部 1 2 3 は、URL対応テーブル 1 2 7 から、ステップ S 2 4 2 で抽出した仮想URLを含むレコードを検索する。

【 0 1 5 7 】

(S 2 4 5) 状態制御部 1 2 3 は、ステップ S 2 4 4 で検索されたレコードに含まれる実URLを削除する。そして、プロキシ状態制御が終了する。

(S 2 4 6) 状態制御部 1 2 3 は、ステップ S 2 4 0 で受信した通知が移動完了通知であるか判断する。受信した通知が移動完了通知である場合はステップ S 2 4 7 に処理が進み、それ以外の場合はプロキシ状態制御が終了する。

【 0 1 5 8 】

(S 2 4 7) 状態制御部 1 2 3 は、受信した移動完了通知に含まれている仮想URL (指定された仮想URL) を抽出する。状態制御部 1 2 3 は、URL対応テーブル 1 2 7 から、抽出した仮想URLを含むレコードを検索する。

【 0 1 5 9 】

(S 2 4 8) 状態制御部 1 2 3 は、受信した移動完了通知に含まれている実URL (指定された実URL) を抽出する。状態制御部 1 2 3 は、ステップ S 2 4 7 で検索されたレコードに、抽出した実URLを登録する。

【 0 1 6 0 】

(S 2 4 9) データ保留部 1 3 4 は、WSハンドシェイク要求としてのHTTPメッセージを生成する。このとき、生成するHTTPメッセージのHTTPヘッダには、端末情報テーブル 1 2 4 に登録されたHTTPヘッダを使用する。

【 0 1 6 1 】

(S 2 5 0) 状態制御部 1 2 3 は、移動管理テーブル 1 2 6 から、ステップ S 2 4 7 で抽出した仮想URLを含むレコードを検索する。

(S 2 5 1) 状態制御部 1 2 3 は、ステップ S 2 5 0 で検索されたレコードに含まれる移動状態をFalseに更新する。

【 0 1 6 2 】

次に、図 1 4 ~ 1 6 に示した処理を、図 8 のシーケンスに沿って説明する。

端末装置 4 1 は、サーバアプリケーション 1 4 0 の仮想URL「ws://vurl:80/app1」が入力されると、「vurl」をDNSサーバ 4 3 に問い合わせ、これに対応するIPアドレス「10.x.x.40」を取得する。その結果、サーバ機能部 1

10

20

30

40

50

31は、端末装置41からの要求に応じてTCPコネクションを確立し、socket1を生成する。そして、サーバ機能部131は、socket1から仮想URL「ws://vurl:80/app1」を指定したWSハンドシェイク要求を受信する。

【0163】

ソケット検索部132は、ソケット対応テーブル125からsocket1を含むレコードを検索する。ここでは、socket1から受信する最初のメッセージであるため、該当するレコードは検索されない。端末情報保存部133は、受信されたWSハンドシェイク要求のHTTPヘッダを、socket1と対応付けて端末情報テーブル124に登録する。データ保留部134は、移動管理テーブル126から、指定された仮想URLを含むレコードを検索し、移動状態がFalseであることを確認する。

10

【0164】

宛先解決部135は、URL対応テーブル127から、指定された仮想URLを含むレコードを検索し、当該仮想URLに対応する実URL「ws://10.x.x.30:8880/app1」を取得する。URL書換部136は、WSハンドシェイク要求に含まれる仮想URLを実URLに書き換える。

【0165】

クライアント機能部137は、socket1を含むレコードがソケット対応テーブル125に存在しないため、実URLが示すサーバアプリケーション140とTCPコネクションを確立し、socket10を生成する。クライアント機能部137は、ソケット対応テーブル125に、socket1とsocket10とを対応付けたレコードを追加する。Upgrade状態はFalseに設定しておく。また、クライアント機能部137は、仮想URLを含む移動管理テーブル126のレコードのインデックス「#1」を、socket1, socket10と対応付けてポインタとして登録する。

20

【0166】

クライアント機能部137は、socket10からWSハンドシェイク要求を送信する。その後、クライアント機能部137は、socket10から、Upgrade=websocketに設定されたWSハンドシェイク応答を受信する。

【0167】

ソケット検索部138は、ソケット対応テーブル125からsocket10を含むレコードを検索する。クローズ無効化部139は、socket10に対応するUpgrade状態がFalseであることを確認する。サーバ機能部131は、受信されたWSハンドシェイク応答のHTTPヘッダにUpgrade=websocketが含まれることを確認する。すると、サーバ機能部131は、socket10に対応するUpgrade状態をTrueに更新する。そして、サーバ機能部131は、socket10と対応付けられたsocket1からWSハンドシェイク応答を送信する。

30

【0168】

サーバ機能部131は、socket1からWebソケットメッセージを受信する。ソケット検索部132は、ソケット対応テーブル125からsocket1を含むレコードを検索し、Upgrade状態がTrueであることを確認する。データ保留部134は、socket1に対応するポインタから移動管理テーブル126のインデックス「#1」のレコードにアクセスし、移動状態がFalseであることを確認する。

40

【0169】

宛先解決部135は、ソケット対応テーブル125からsocket1と対応付けられたsocket10を取得する。クライアント機能部137は、socket10からWebソケットメッセージを送信する。その後、クライアント機能部137は、socket10からWebソケットメッセージを受信する。

【0170】

ソケット検索部138は、ソケット対応テーブル125からsocket10を含むレコードを検索する。クローズ無効化部139は、socket10に対応するUpgrade状態がTrueであることを確認する。すると、クローズ無効化部139は、soc

50

socket 10に対応するポインタから移動管理テーブル126のインデックス「#1」のレコードにアクセスし、移動状態がFalseであることを確認する。この場合、クローズ無効化部139は、クローズ通知の終端は不要であると判断する。サーバ機能部131は、socket 10と対応付けられたsocket 1からWSソケットメッセージを送信する。

【0171】

サーバアプリケーション140の移動が決定されると、移動開始通知部232は、仮想URL「ws://vurl:80/app1」を指定した移動開始通知をリバースプロキシ120に送信する。状態制御部123は、移動管理テーブル126から当該仮想URLを含むレコードを検索し、当該仮想URLに対応する移動状態をTrueに更新する。また、状態制御部123は、URL対応テーブル127から当該仮想URLを含むレコードを検索し、当該仮想URLに対応する実URLを削除する(NULLに更新する)。

10

【0172】

配備計画サーバ200は、Webサーバ100a上のサーバアプリケーション140に停止指示を送信する。すると、サーバアプリケーション140は、Webソケットのクローズ通知を各TCPコネクション上で送信する。クライアント機能部137は、socket 10からクローズ通知を受信する。

【0173】

ソケット検索部138は、ソケット対応テーブル125からsocket 10を含むレコードを検索する。クローズ無効化部139は、socket 10に対応するUpgrade状態がTrueであることを確認する。すると、クローズ無効化部139は、socket 10に対応するポインタから移動管理テーブル126のインデックス「#1」のレコードにアクセスし、移動状態がTrueであることを確認する。すると、クローズ無効化部139は、クローズ通知を破棄してサーバ機能部131に出力しない。

20

【0174】

サーバ機能部131は、ソケット対応テーブル125からsocket 10を削除する。クライアント機能部137は、サーバアプリケーション140との間のTCPコネクションを切断し、socket 10を削除する。

【0175】

端末装置41とリバースプロキシ120との間のTCPコネクションは切断されていないため、サーバ機能部131は、socket 1からWebソケットメッセージを受信する。ソケット検索部132は、ソケット対応テーブル125からsocket 1を含むレコードを検索し、Upgrade状態がTrueであることを確認する。データ保留部134は、socket 1に対応するポインタから移動管理テーブル126のインデックス「#1」のレコードにアクセスし、移動状態がTrueであることを確認する。すると、データ保留部134は、データ記憶部122にWebソケットメッセージを保存する。

30

【0176】

サーバアプリケーション140がWebサーバ100aからWebサーバ100に移動すると、移動完了通知部233は、リバースプロキシ120に移動完了通知を送信する。移動完了通知には、仮想URL「ws://vurl:80/app1」と実URL「ws://10.x.x.40:8880/app1」とが含まれる。

40

【0177】

状態制御部123は、URL対応テーブル127から当該仮想URLを含むレコードを検索し、当該仮想URLと対応付けて通知された実URLを登録する。また、状態制御部123は、移動管理テーブル126から当該仮想URLを含むレコードを検索し、当該仮想URLに対応する移動状態をFalseに更新する。

【0178】

データ保留部134は、移動状態がTrueからFalseに変化すると、データ記憶部122に記憶されたWebソケットメッセージを転送するため、WSハンドシェイク要求を生成する。このとき、データ保留部134は、WSハンドシェイク要求のHTTPヘ

50

ッダに、端末情報テーブル124に登録されたHTTPヘッダを使用する。データ保留部134は、WSハンドシェイク要求に続いて、データ記憶部122に記憶されたWebソケットメッセージを出力する。なお、データ保留部134が移動管理テーブル126に登録された移動状態を監視する代わりに、移動状態がFalseに変わったことを状態制御部123がデータ保留部134に通知するようにしてもよい。

【0179】

クライアント機能部137は、socket1に対応するサーバ側ソケットがソケット対応テーブル125に登録されていないため、実URLが示すサーバアプリケーション140とTCPコネクションを確立し、socket11を生成する。クライアント機能部137は、ソケット対応テーブル125に、socket1と対応付けてsocket11を登録する。クライアント機能部137は、データ保留部134が生成したWSハンドシェイク要求をsocket11から送信し、WSハンドシェイク応答をsocket11から受信する。その後、クライアント機能部137は、データ記憶部122から取り出されたWebソケットメッセージをsocket11から送信する。

10

【0180】

第2の実施の形態の情報処理システムによれば、サーバアプリケーション140が移動するとき、サーバアプリケーション140が発行したクローズ通知がリバースプロキシ120、120bによって終端される。これにより、端末装置41、42、51、52に対してWSセッションが継続していると認識させることができ、クローズ通知に反応して端末装置41、42、51、52がTCPコネクションを切断してしまうのを抑制できる。

20

【0181】

そして、移動完了後は、リバースプロキシ120、120bによって、端末装置41、42、51、52側のTCPコネクションとサーバアプリケーション140側のTCPコネクションとが対応付けられ、両者の間でWebソケットメッセージが転送される。これにより、端末装置41、42、51、52は、サーバアプリケーション140が移動する前に確立したTCPコネクションを移動後も使用し続けることができ、TCPコネクションを確立し直さなくてよい。よって、端末装置41、42、51、52のユーザは、ログイン手続などの操作を行わずに移動後もサーバアプリケーション140のサービスを継続して利用することができ、ユーザの利便性が低下しないようにすることができる。

【0182】

また、サーバアプリケーション140の移動時にはWSセッションのデータも移動するため、サーバアプリケーション140は移動前の通信のコンテキストを引き継ぐことができ、WSセッションを継続することができる。また、リバースプロキシ120、120bは、サーバアプリケーション140が移動中であるときはクローズ通知を終端する一方、サーバアプリケーション140が移動中でないときはクローズ通知を終端せずに転送する。これにより、通常時のWebソケット通信を正常に行うことができる。

30

【0183】

[第3の実施の形態]

次に、第3の実施の形態を説明する。第2の実施の形態との違いを中心に説明し、第2の実施の形態と同様の事項については説明を省略することができる。

40

【0184】

第2の実施の形態では、サーバアプリケーションを移動した後に、遠隔のWebサーバの間でTCPコネクションを確立した。これに対し、第3の実施の形態では、サーバアプリケーションを移動する前に、予め遠隔のWebサーバの間でTCPコネクションを確立しておく。事前のTCPコネクションは、異なるWebサーバ上のリバースプロキシの間で確立しておく。これにより、サーバアプリケーション140のサービスが実質的に中断される時間を短縮することができる。

【0185】

図17は、第3の実施の形態のサーバアプリケーションの配備例を示す図である。

第3の実施の形態の情報処理システムは、第2の実施の形態のWebサーバ100、1

50

00a, 100bに代えて、Webサーバ100c, 100d, 100eを含む。Webサーバ100cは、拠点40に設置されている。Webサーバ100dは、クラウド30に設置されている。Webサーバ100eは、拠点50に設置されている。

【0186】

Webサーバ100cは、リバースプロキシ120cおよび実行環境150cを実行する。Webサーバ100dは、リバースプロキシ120dおよび実行環境150dを実行する。Webサーバ100eは、リバースプロキシ120eおよび実行環境150eを実行する。サーバアプリケーション140が、最初はWebサーバ100dで実行されており、その後Webサーバ100eに移動する場合を考える。

【0187】

Webサーバ100cのIPアドレスは「10.x.x.40」である。Webサーバ100dのIPアドレスは「10.x.x.30」である。Webサーバ100eのIPアドレスは「10.x.x.50」である。サーバアプリケーション140がWebサーバ100dに配置されている場合、サーバアプリケーション140には「ws://10.x.x.30:8880」を含む実URLが付与される。サーバアプリケーション140がWebサーバ100eに配置されている場合、サーバアプリケーション140には「ws://10.x.x.50:8880」を含む実URLが付与される。

【0188】

リバースプロキシ120c, 120eには「ws://vurl:80」を含む仮想URLが付与される。また、リバースプロキシ120cには、仮想URLとは別に、「ws://10.x.x.40:80」を含む実URLが付与されている。リバースプロキシ120eには、仮想URLとは別に、「ws://10.x.x.50:80」を含む実URLが付与されている。リバースプロキシ120dには、「ws://10.x.x.30:80」を含む実URLが付与されている。第3の実施の形態では、リバースプロキシ120c, 120eは、他のWebサーバに配置されたサーバアプリケーションとは直接通信せず、他のリバースプロキシを経由して通信する。リバースプロキシ120c, 120d, 120eの実URLは、リバースプロキシ間の通信に用いられる。

【0189】

サーバアプリケーション140がWebサーバ100dに配置されている場合、リバースプロキシ120cは、仮想URLを含むアクセスを受信すると、仮想URLをリバースプロキシ120dの実URLに変換し、アクセスをリバースプロキシ120dに転送する。リバースプロキシ120dは、リバースプロキシ120dの実URLをサーバアプリケーション140の実URLに変換し、アクセスをサーバアプリケーション140に転送する。また、サーバアプリケーション140がWebサーバ100eに配置されている場合、リバースプロキシ120cは、仮想URLをリバースプロキシ120eの実URLに変換し、アクセスをリバースプロキシ120eに転送する。リバースプロキシ120eは、リバースプロキシ120eの実URLをサーバアプリケーション140の実URLに変換し、アクセスをサーバアプリケーション140に転送する。

【0190】

図18は、Webソケットによる第4の通信例を示すシーケンス図である。

端末装置41は、リバースプロキシ120cとTCPハンドシェイクを行い、端末装置41とリバースプロキシ120cとの間にTCPコネクションを確立する(S170)。端末装置41は、サーバアプリケーション140の仮想URLを指定したWSハンドシェイク要求をリバースプロキシ120cに送信する(S171)。リバースプロキシ120cは、仮想URLをリバースプロキシ120dの実URLに変換する。

【0191】

リバースプロキシ120cは、リバースプロキシ120dとTCPハンドシェイクを行い、リバースプロキシ120cとリバースプロキシ120dとの間にTCPコネクションを確立する。リバースプロキシ120dは、サーバアプリケーション140とTCPハンドシェイクを行い、リバースプロキシ120dとサーバアプリケーション140との間に

10

20

30

40

50

T C Pコネクションを確立する (S 1 7 2)。

【 0 1 9 2 】

リバースプロキシ 1 2 0 c は、W S ハンドシェイク要求をリバースプロキシ 1 2 0 d に転送する。リバースプロキシ 1 2 0 d は、リバースプロキシ 1 2 0 d の実 U R L をサーバアプリケーション 1 4 0 の実 U R L に変換し、W S ハンドシェイク要求をサーバアプリケーション 1 4 0 に転送する (S 1 7 3)。サーバアプリケーション 1 4 0 は、W S ハンドシェイク応答をリバースプロキシ 1 2 0 d に送信する。リバースプロキシ 1 2 0 d は、W S ハンドシェイク応答をリバースプロキシ 1 2 0 c に転送する。リバースプロキシ 1 2 0 c は、W S ハンドシェイク応答を端末装置 4 1 に転送する (S 1 7 4)。

【 0 1 9 3 】

端末装置 4 1 は、W e b ソケットメッセージをリバースプロキシ 1 2 0 c に送信する。リバースプロキシ 1 2 0 c は、W e b ソケットメッセージをリバースプロキシ 1 2 0 d に転送する。リバースプロキシ 1 2 0 d は、W e b ソケットメッセージをサーバアプリケーション 1 4 0 に転送する (S 1 7 5)。

【 0 1 9 4 】

ここで、配備計画サーバ 2 0 0 a (後述する) が、サーバアプリケーション 1 4 0 を W e b サーバ 1 0 0 d から W e b サーバ 1 0 0 e に移動すると決定したとする。配備計画サーバ 2 0 0 a がリバースプロキシ 1 2 0 c に移動準備通知を送信すると、リバースプロキシ 1 2 0 c は事前に、移動先の W e b サーバ 1 0 0 e のリバースプロキシ 1 2 0 e と T C P ハンドシェイクを行い、T C P コネクションを確立しておく (S 1 7 6)。

【 0 1 9 5 】

その後、配備計画サーバ 2 0 0 a がサーバアプリケーション 1 4 0 に移動指示を送信すると、サーバアプリケーション 1 4 0 はクローズ通知をリバースプロキシ 1 2 0 d に送信する。リバースプロキシ 1 2 0 d は、クローズ通知をリバースプロキシ 1 2 0 c に転送する。リバースプロキシ 1 2 0 c は、受信したクローズ通知を終端する (S 1 7 7)。

【 0 1 9 6 】

リバースプロキシ 1 2 0 c は、クローズ通知を受信すると、リバースプロキシ 1 2 0 c とリバースプロキシ 1 2 0 d との間の T C P コネクションを切断する。また、リバースプロキシ 1 2 0 d は、クローズ通知を受信すると、リバースプロキシ 1 2 0 d とサーバアプリケーション 1 4 0 との間の T C P コネクションを切断する (S 1 7 8)。

【 0 1 9 7 】

図 1 9 は、W e b ソケットによる第 4 の通信例を示すシーケンス図 (続き) である。

端末装置 4 1 は、W S セッションが継続していると認識しているため、W e b ソケットメッセージをリバースプロキシ 1 2 0 c に送信する (S 1 7 9)。リバースプロキシ 1 2 0 c は、リバースプロキシ 1 2 0 c とリバースプロキシ 1 2 0 e との間に T C P コネクションが存在しないため、受信した W e b ソケットメッセージをバッファに保存しておく。

【 0 1 9 8 】

そして、リバースプロキシ 1 2 0 c は、配備計画サーバ 2 0 0 a から移動完了通知を受信すると、W S ハンドシェイク要求を生成してリバースプロキシ 1 2 0 e に送信する (S 1 8 0)。この W S ハンドシェイク要求の送信には、上記ステップ S 1 7 6 で事前に確立した T C P コネクションが用いられる。よって、移動完了通知を受けてから W e b サーバ 1 0 0 c と W e b サーバ 1 0 0 e との間で T C P ハンドシェイクを行わなくてよい。

【 0 1 9 9 】

リバースプロキシ 1 2 0 e は、サーバアプリケーション 1 4 0 と T C P ハンドシェイクを行い、リバースプロキシ 1 2 0 e とサーバアプリケーション 1 4 0 との間に T C P コネクションを確立する (S 1 8 1)。この T C P コネクションは、W e b サーバ 1 0 0 e の内部に形成されるコネクションである。よって、ステップ S 1 8 1 の T C P ハンドシェイクのコストは、ステップ S 1 7 6 の T C P ハンドシェイクのコストより十分に小さい。

【 0 2 0 0 】

リバースプロキシ 1 2 0 e は、リバースプロキシ 1 2 0 c から受信した W S ハンドシェ

10

20

30

40

50

イク要求をサーバアプリケーション140に転送する(S182)。サーバアプリケーション140は、WSハンドシェイク応答をリバースプロキシ120eに送信する。リバースプロキシ120eは、WSハンドシェイク応答をリバースプロキシ120cに転送する(S183)。リバースプロキシ120cは、バッファに保存されたWebソケットメッセージをリバースプロキシ120eに転送する。リバースプロキシ120eは、Webソケットメッセージをサーバアプリケーション140に転送する(S184)。

【0201】

正常にWSセッションを終了する場合、サーバアプリケーション140は、クローズ通知をリバースプロキシ120eに送信する。リバースプロキシ120eは、クローズ通知をリバースプロキシ120cに転送する。リバースプロキシ120cは、クローズ通知を端末装置41に転送する(S185)。端末装置41は、端末装置41とリバースプロキシ120cとの間のTCPコネクションを切断する。リバースプロキシ120cは、リバースプロキシ120cとリバースプロキシ120eとの間のTCPコネクションを切断する。リバースプロキシ120eは、リバースプロキシ120eとサーバアプリケーション140との間のTCPコネクションを切断する(S186)。

10

【0202】

サーバアプリケーション140から見て、ステップS172からステップS178まで1つのTCPコネクションが維持され、ステップS181からステップS186まで1つのTCPコネクションが維持されている。サーバアプリケーション140から見て、ステップS178とステップS181の間はTCPコネクションが存在しない。

20

【0203】

これに対し、端末装置41から見て、ステップS170からステップS186までTCPコネクションが切断されずに維持されている。また、端末装置41からは、ステップS174からステップS185までWSセッションが継続しているように見える。また、サーバアプリケーション140の移動の際にはセッションデータも移動する。よって、端末装置41は、サーバアプリケーション140の移動を意識せずに通信を継続できる。

【0204】

図20は、第3の実施の形態の移動制御の通信例を示すシーケンス図である。

配備計画サーバ200aは、サーバアプリケーション140をWebサーバ100dからWebサーバ100eに移動することを決定する。すると、配備計画サーバ200aは、Webサーバ100cのリバースプロキシ120cとWebサーバ100eのリバースプロキシ120eに、移動準備通知を送信する(S190)。移動準備通知は、サーバアプリケーション140を識別する仮想URLと、移動先のWebサーバ100eに配置されたリバースプロキシ120eの実URLとを含む。

30

【0205】

リバースプロキシ120cは、サーバアプリケーション140に関して確立されているTCPコネクション1つにつき、新たなTCPコネクション1つをリバースプロキシ120eと確立する。異なるWebサーバ間のTCPハンドシェイクが終わると、リバースプロキシ120c、120eは、配備計画サーバ200aに応答を送信する(S191)。

【0206】

移動準備通知に対する応答を全てのリバースプロキシから受信すると、配備計画サーバ200aは、リバースプロキシ120c、120eに移動開始通知を送信する(S192)。移動開始通知は、仮想URLを含む。リバースプロキシ120c、120eは、配備計画サーバ200aに応答を送信する(S193)。

40

【0207】

配備計画サーバ200aは、サーバアプリケーション140に停止指示を送信する。サーバアプリケーション140は、Webソケットのために維持しているTCPコネクション毎にクローズ通知を送信する(S194)。例えば、サーバアプリケーション140は、リバースプロキシ120c、120eに1または2以上のクローズ通知を送信する。このクローズ通知は、図18のステップS177のクローズ通知に対応する。

50

【 0 2 0 8 】

配備計画サーバ200aは、Webサーバ100dの実行環境150dに移動指示を送信する。実行環境150dは、サーバアプリケーション140のプログラムとセッションデータをWebサーバ100eに送信する(S195)。実行環境150dは、サーバアプリケーション140を停止させる。Webサーバ100eの実行環境150eは、プログラムとセッションデータによりサーバアプリケーション140を起動する。

【 0 2 0 9 】

配備計画サーバ200aは、リバースプロキシ120c, 120eに移動完了通知を送信する(S196)。移動完了通知は、仮想URLと、移動先のWebサーバ100eに配置されたリバースプロキシ120eの実URLとを含む(S196)。

10

【 0 2 1 0 】

図21は、第3の実施の形態のリバースプロキシの機能例を示すブロック図である。

リバースプロキシ120cは、制御情報記憶部121、データ記憶部122、状態制御部123およびコネクションプール部128を有する。また、リバースプロキシ120cは、サーバ機能部131、ソケット検索部132、端末情報保存部133、データ保留部134、宛先解決部135、URL書換部136、クライアント機能部137c、ソケット検索部138およびクローズ無効化部139を有する。リバースプロキシ120d, 120eも、リバースプロキシ120cと同様のモジュールにより実装できる。

【 0 2 1 1 】

制御情報記憶部121は、第2の実施の形態と同様に、端末情報テーブル124、ソケット対応テーブル125、移動管理テーブル126およびURL対応テーブル127を有する。ただし、ソケット対応テーブル125のサーバ側ソケットの項目には、他のリバースプロキシとの間に確立したTCPコネクションに対応するソケットの番号が登録されることがある。また、URL対応テーブル127の実URLの項目には、他のリバースプロキシの実URLが登録されることがある。

20

【 0 2 1 2 】

コネクションプール部128は、配備計画サーバ200aから移動準備通知を受信すると、移動管理テーブル126から、指定された仮想URLを含むレコードを検索し、検索されたレコードのインデックスを取得する。コネクションプール部128は、ソケット対応テーブル125から、取得したインデックスを含むレコードを検索し、検索されたレコードをカウントする。すなわち、コネクションプール部128は、指定された仮想URLに関して、現在存在するTCPコネクションをカウントする。そして、コネクションプール部128は、移動準備通知に含まれる実URLが示すリバースプロキシとの間に、カウントした数のTCPコネクションを確立してプールしておく。

30

【 0 2 1 3 】

クライアント機能部137cは、サーバアプリケーション140が移動した直後に他のリバースプロキシと通信するときは、新たなTCPコネクションを確立する代わりにプールされているTCPコネクションを選択して使用する。クライアント機能部137cは、選択したTCPコネクションに対応するソケットの番号を、サーバ機能部131側のソケットと対応付けてソケット対応テーブル125に登録する。

40

【 0 2 1 4 】

配備計画サーバ200aは、制御情報記憶部221、移動制御部231、移動開始通知部232、移動完了通知部233aおよび移動準備通知部234を有する。

移動完了通知部233aは、サーバアプリケーション140の移動が完了すると、リバースプロキシ120c, 120eに対して移動完了通知を送信する。移動完了通知は、サーバアプリケーション140を識別する仮想URLと、移動先のWebサーバのリバースプロキシに付与された実URLとが含まれる。

【 0 2 1 5 】

移動準備通知部234は、サーバアプリケーション140の移動が決定されると、移動開始通知に先立って、リバースプロキシ120c, 120eに対して移動準備通知を送信

50

する。移動準備通知は、サーバアプリケーション140を識別する仮想URLと、移動先のWebサーバのリバースプロキシに付与された実URLとが含まれる。

【0216】

なお、サーバアプリケーション140の移動前に新たなTCPコネクションを確立すると、一時的に、古いTCPコネクションと新たなTCPコネクションとが併存し、第2の実施の形態の方法の2倍のTCPコネクションを維持することになる。よって、Webサーバ100c, 100eの負荷が過大になるおそれがある。そこで、Webサーバ100c, 100eの負荷を監視し、負荷に応じて、サーバアプリケーション140の移動前に新たなTCPコネクションを確立するか、サーバアプリケーション140の移動後に新たなTCPコネクションを確立するかを選択するようにしてもよい。

10

【0217】

例えば、配備計画サーバ200aは、Webサーバ100c, 100eのメモリ使用量を監視する。Webサーバ100c, 100eの空きメモリ量が閾値以上ある場合、配備計画サーバ200aは、Webサーバ100c, 100eに移動準備通知を送信する。これにより、サーバアプリケーション140の移動前に新たなTCPコネクションが確立される。一方、Webサーバ100c, 100eの空きメモリ量が閾値未満である場合、配備計画サーバ200aは、Webサーバ100c, 100eに移動準備通知を送信しない。これにより、サーバアプリケーション140の移動後に新たなTCPコネクションが確立される。ただし、配備計画サーバ200aは、Webサーバ100c, 100eのCPU使用率を監視し、CPU使用率が閾値以下の場合には移動準備通知を送信し、CPU使用率が閾値を超える場合は移動準備通知を送信しないようにしてもよい。また、判断基準として、CPU使用率とメモリ使用量の両方を用いてもよい。

20

【0218】

第3の実施の形態の情報処理システムによれば、第2の実施の形態の情報処理システムと同様の効果が得られる。更に、第3の実施の形態の情報処理システムによれば、サーバアプリケーション140がクローズ通知を送信してから再びサーバアプリケーション140にWebソケットメッセージを転送可能になるまでの時間を短縮できる。よって、端末装置41, 42, 51, 52から見て実質的にサーバアプリケーション140のサービスが中断されている時間を短縮でき、ユーザの利便性を向上させることができる。

30

【0219】

なお、前述のように、第1の実施の形態の情報処理は、コンピュータにプログラムを実行させることで実現できる。第2の実施の形態の情報処理は、Webサーバ100, 100a, 100bおよび配備計画サーバ200にプログラムを実行させることで実現できる。第3の実施の形態の情報処理は、Webサーバ100c, 100d, 100eおよび配備計画サーバ200aにプログラムを実行させることで実現できる。

【0220】

プログラムは、コンピュータ読み取り可能な記録媒体（例えば、記録媒体113）に記録しておくことができる。記録媒体として、例えば、磁気ディスク、光ディスク、光磁気ディスク、半導体メモリなどを使用できる。磁気ディスクには、FDおよびHDDが含まれる。光ディスクには、CD、CD-R（Recordable）/RW（Rewritable）、DVDおよびDVD-R/RWが含まれる。プログラムは、可搬型の記録媒体に記録されて配布されることがある。その場合、可搬型の記録媒体から他の記録媒体（例えば、HDD103）にプログラムをコピーして実行してもよい。

40

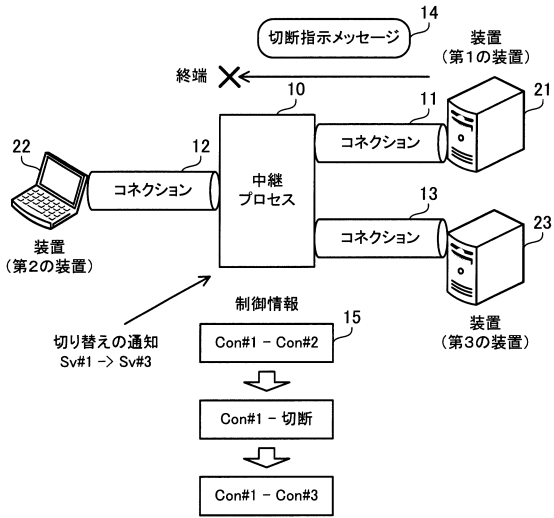
【符号の説明】

【0221】

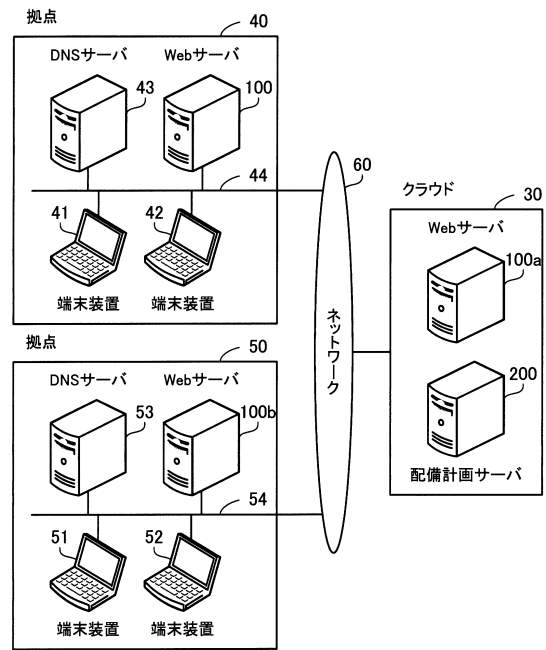
- 10 中継プロセス
- 11, 12, 13 コネクション
- 14 切断指示メッセージ
- 15 制御情報
- 21, 22, 23 装置

50

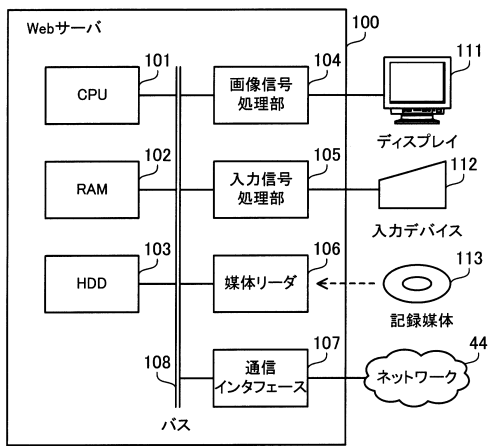
【図1】



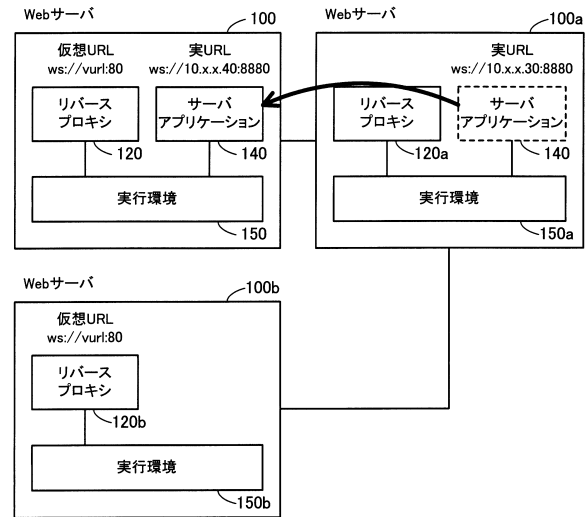
【図2】



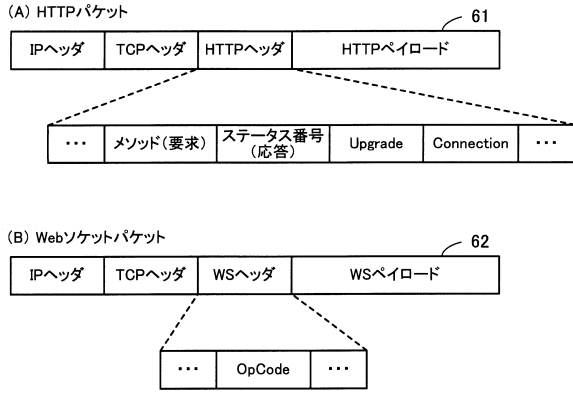
【図3】



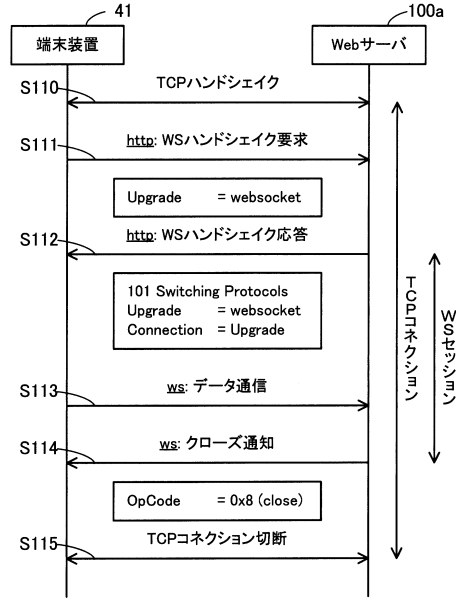
【図4】



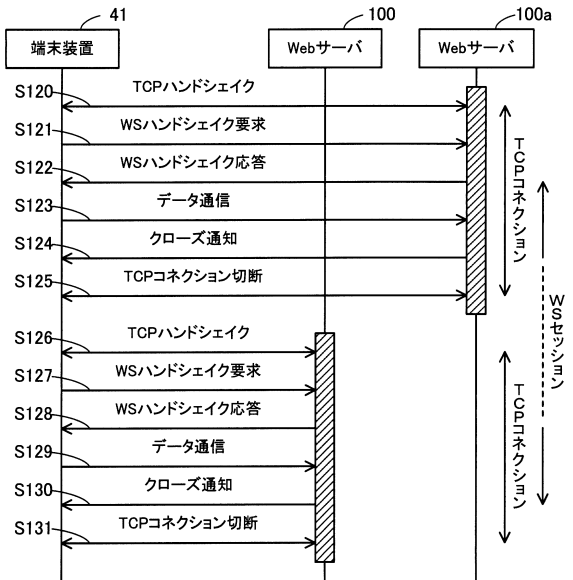
【図5】



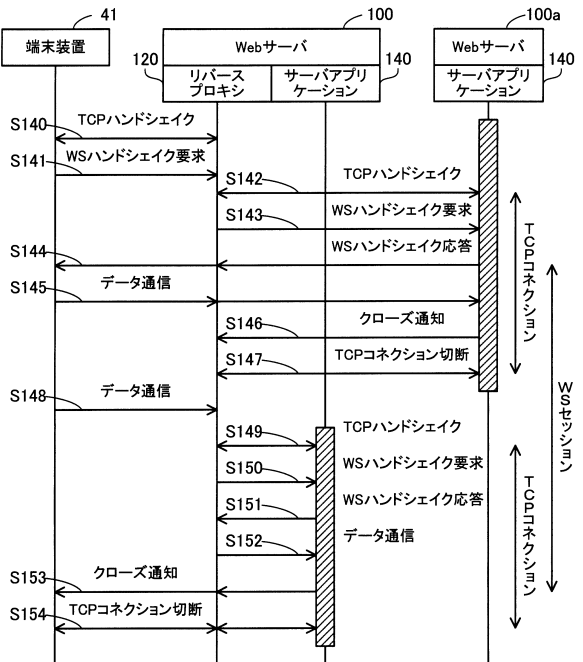
【図6】



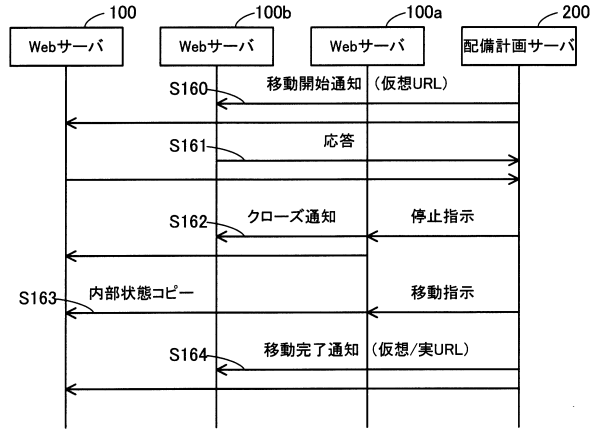
【図7】



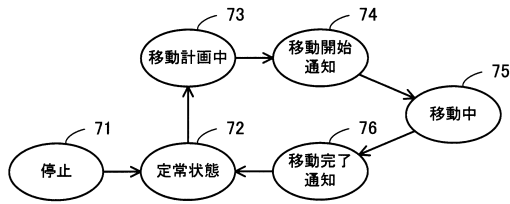
【図8】



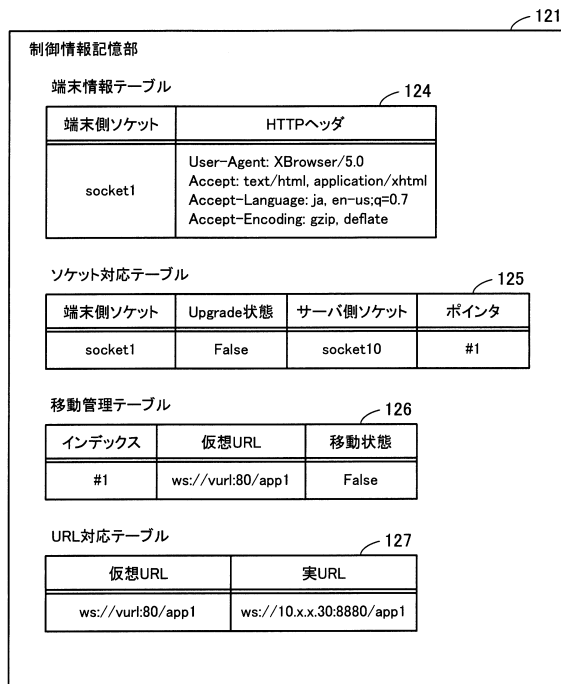
【図9】



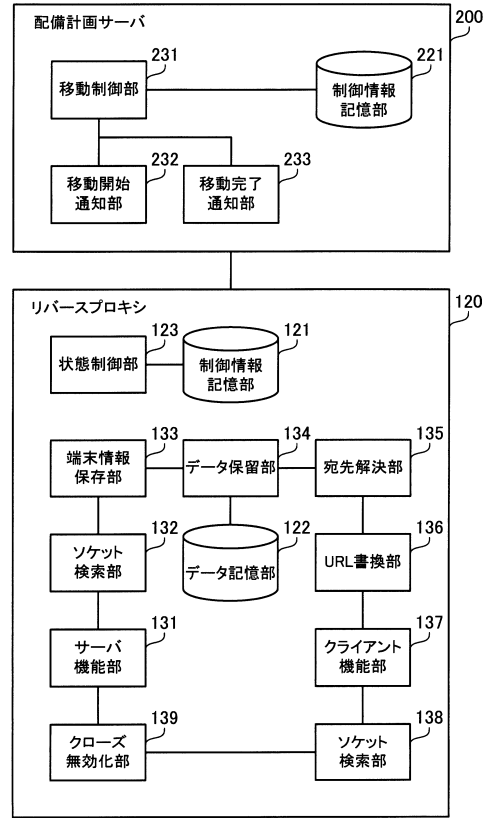
【図10】



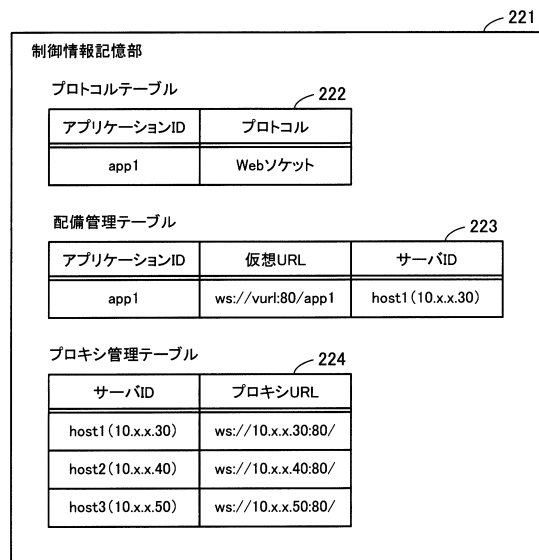
【図12】



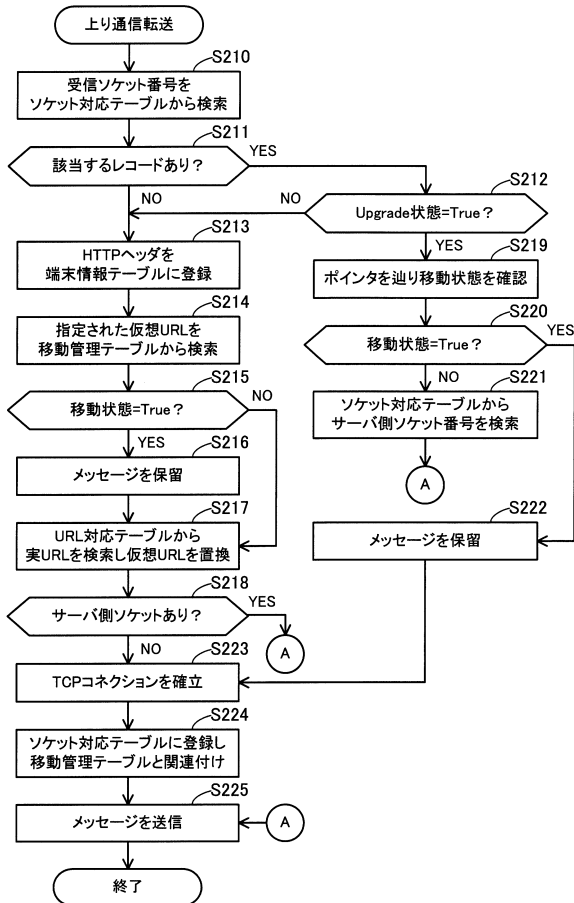
【図11】



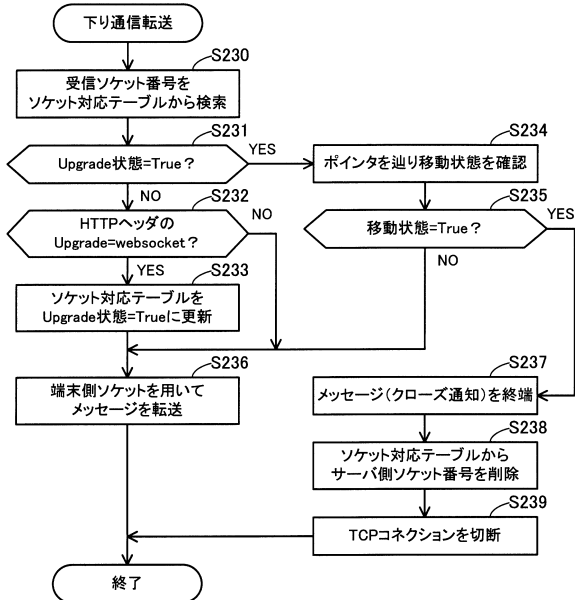
【図13】



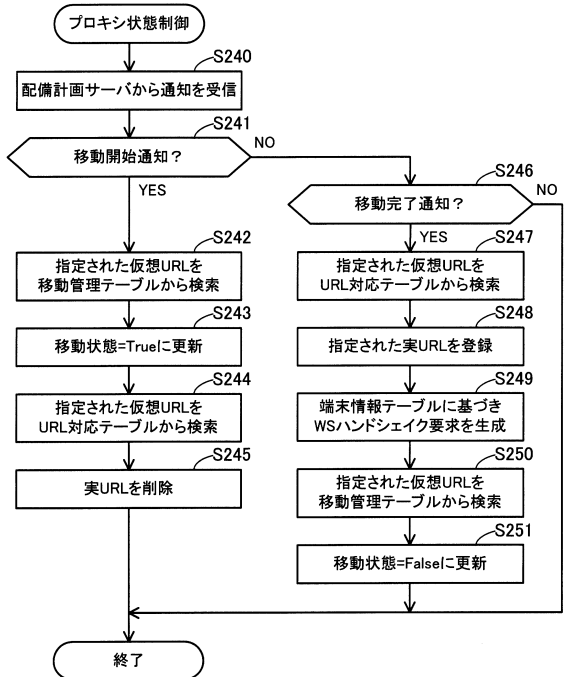
【図14】



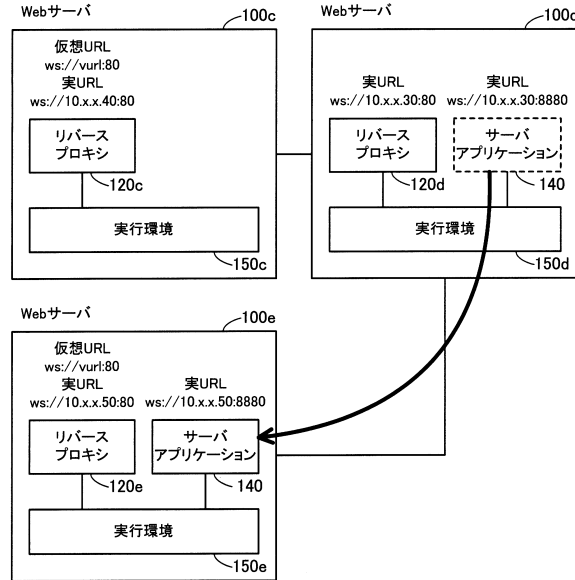
【図15】



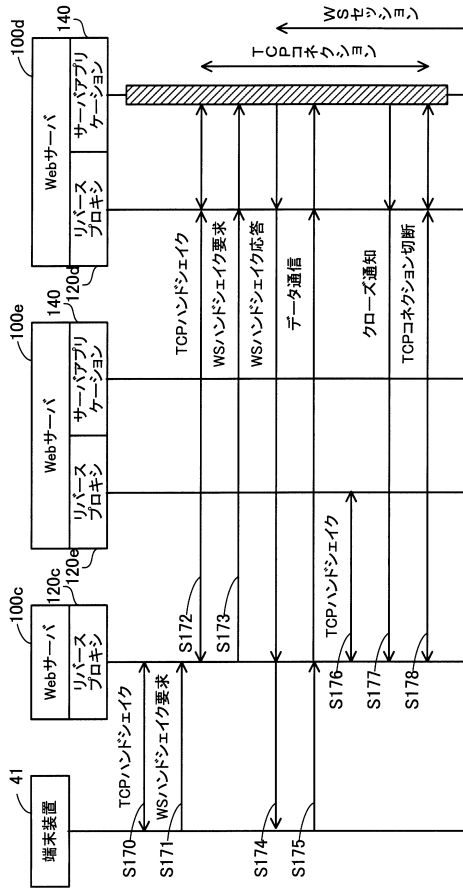
【図16】



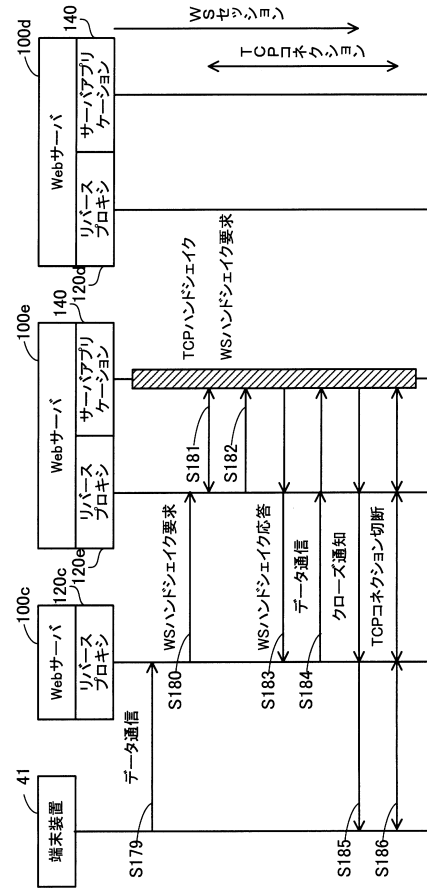
【図17】



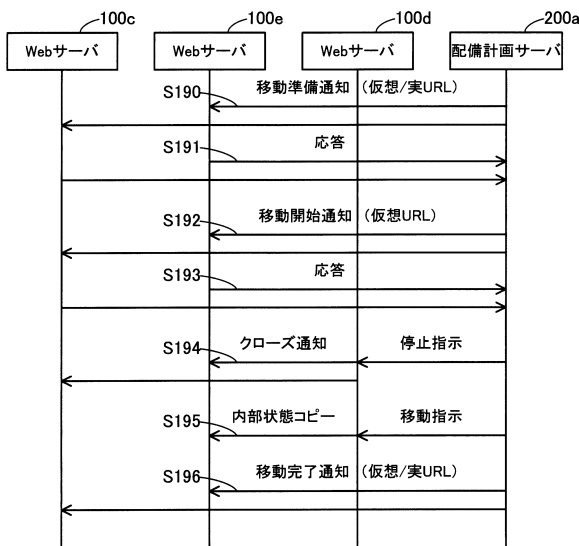
【図18】



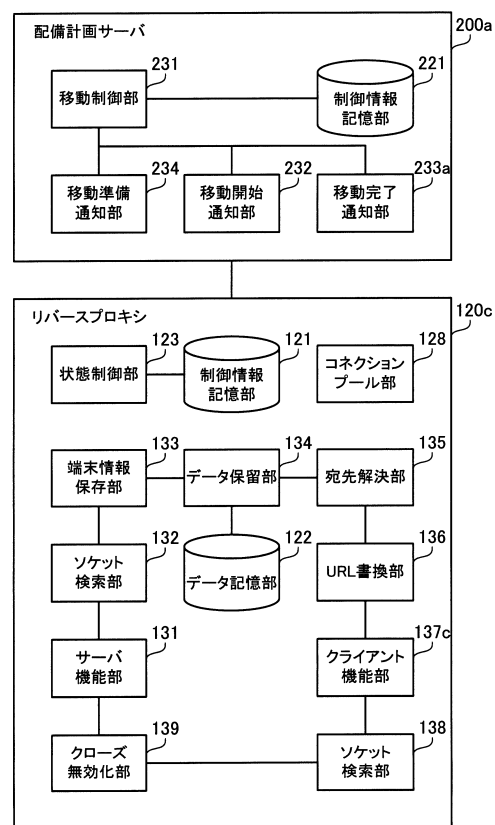
【図19】



【図20】



【図21】



フロントページの続き

- (56)参考文献 特開2006-215635(JP,A)
米国特許出願公開第2002/0169889(US,A1)
特開2001-117899(JP,A)
特開2007-272472(JP,A)
特開2004-192233(JP,A)

(58)調査した分野(Int.Cl., DB名)

G06F 13/00
H04L 12/70
H04L 29/08