(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第6346690号
(P6346690)
(45)発行日　平成30年6月20日(2018.6.20)　(24)登録日　平成30年6月1日(2018.6.1)

(51)Int.Cl.　F I
H04L 12/70 (2013.01)　H04L 12/70 100Z

請求項の数 20　外国語出願　(全 30 頁)

(21)出願番号　特願2017-75430(P2017-75430)
(22)出願日　平成29年4月5日(2017.4.5)
(62)分割の表示　特願2016-42758(P2016-42758)の分割
原出願日　平成24年11月15日(2012.11.15)
(65)公開番号　特開2017-153118(P2017-153118A)
(43)公開日　平成29年8月31日(2017.8.31)
審査請求日　平成29年4月20日(2017.4.20)
(31)優先権主張番号　61/560,279
(32)優先日　平成23年11月15日(2011.11.15)
(33)優先権主張国　米国(US)

(73)特許権者　511235548
ニシラ, インコーポレイテッド
アメリカ合衆国 カリフォルニア州 ９４３０４, パロアルト, ヒルビュー アベニュー ３４０１
(74)代理人　100076428
弁理士 大塚 康徳
(74)代理人　100115071
弁理士 大塚 康弘
(74)代理人　100112508
弁理士 高柳 司郎
(74)代理人　100116894
弁理士 木村 秀二
(74)代理人　100130409
弁理士 下山 治

最終頁に続く

(54)【発明の名称】ミドルボックスを構成設定するネットワーク制御システム

(57)【特許請求の範囲】
【請求項１】
複数の論理ネットワークに属する複数のエンドマシンのホストとなるホストマシンで動作するミドルボックス要素に関する方法であって、
前記ホストマシンで動作する管理された転送要素からデータパケットを受信することと、
前記管理された転送要素により前記データパケットに付加されたタグに基づいて、複数の論理ネットワークのために、前記ミドルボックス要素により実装される複数の論理ミドルボックスのうちの１つの論理ミドルボックスを識別することと、
前記識別された論理ミドルボックスに関する構成設定に従って前記データパケットを処理することと、
前記処理されたデータパケットを前記管理された転送要素に送信することを有する方法。
【請求項２】
前記データパケットは、前記ホストマシンによってホストされる前記複数のエンドマシンのうちの１つのエンドマシンにより前記管理された転送要素に送信される請求項１に記載の方法。
【請求項３】
前記エンドマシンは、前記識別された論理ミドルボックスと同じ論理ネットワークに属する請求項２に記載の方法。

【請求項４】
前記ミドルボックス要素は、前記管理された転送要素と前記ミドルボックス要素との間でネゴシエートされたソフトウェアポートを介して前記管理された転送要素から前記データパケットを受信する請求項１に記載の方法。
【請求項５】
前記タグは、前記管理された転送要素により前記データパケットの先頭に付加される請求項１に記載の方法。
【請求項６】
前記論理ミドルボックスを識別することは、バインディングテーブルを用いて、前記タグを前記論理ミドルボックスにマッピングすることを有する請求項１に記載の方法。
【請求項７】
前記処理されたデータパケットは、前記タグとともに前記管理された転送要素に送信される請求項１に記載の方法。
【請求項８】
前記管理された転送要素は、前記データパケットの宛先ネットワークのアドレス以外のデータに基づいて前記データパケットをルーティングするルーティングポリシーに従って、前記データパケットを前記ミドルボックス要素に送信する請求項１に記載の方法。
【請求項９】
前記データは、前記データパケットが受信された発信元ポートである請求項８に記載の方法。
【請求項１０】
前記管理された転送要素は、前記処理されたデータパケットを受信し、その後、前記処理されたデータパケットの前記宛先ネットワークのアドレスに基づいて、前記処理されたデータパケットをルーティングする請求項８に記載の方法。
【請求項１１】
複数の論理ネットワークに属する複数のエンドマシンのホストとなるホストマシンの少なくとも１つの処理ユニットで実行されるミドルボックス要素であって、前記ミドルボックス要素は、
前記ホストマシンで動作する管理された転送要素からデータパケットを受信することと、
前記管理された転送要素により前記データパケットに付加されたタグに基づいて、複数の論理ネットワークのために、前記ミドルボックス要素により実装される複数の論理ミドルボックスのうちの１つの論理ミドルボックスを識別することと、
前記識別された論理ミドルボックスに関する構成設定に従って前記データパケットを処理することと、
前記処理されたデータパケットを前記管理された転送要素に送信することと、のための命令のセットを有するミドルボックス要素。
【請求項１２】
前記データパケットは、前記ホストマシンによってホストされる前記複数のエンドマシンのうちの１つのエンドマシンにより前記管理された転送要素に送信される請求項１１に記載のミドルボックス要素。
【請求項１３】
前記エンドマシンは、前記識別された論理ミドルボックスと同じ論理ネットワークに属する請求項１２に記載のミドルボックス要素。
【請求項１４】
前記データパケットは、前記管理された転送要素と前記ミドルボックス要素との間でネゴシエートされたソフトウェアポートを介して前記管理された転送要素から受信される請求項１１に記載のミドルボックス要素。
【請求項１５】
前記タグは、前記管理された転送要素により前記データパケットの先頭に付加される請

求項１１に記載のミドルボックス要素。
【請求項１６】
前記論理ミドルボックスを識別するための命令のセットは、バインディングテーブルを用いて、前記タグを前記論理ミドルボックスにマッピングする命令セットを有する請求項１１に記載のミドルボックス要素。
【請求項１７】
前記処理されたデータパケットは、前記タグとともに前記管理された転送要素に送信される請求項１１に記載のミドルボックス要素。
【請求項１８】
前記管理された転送要素は、前記データパケットの宛先ネットワークのアドレス以外のデータに基づいて前記データパケットをルーティングするルーティングポリシーに従って、前記データパケットを前記ミドルボックス要素に送信する請求項１１に記載のミドルボックス要素。
【請求項１９】
前記データは、前記データパケットが受信された発信元ポートであり、前記管理された転送要素は、前記処理されたデータパケットを受信し、その後、前記処理されたデータパケットの前記宛先ネットワークのアドレスに基づいて、前記処理されたデータパケットをルーティングする請求項１８に記載のミドルボックス要素。
【請求項２０】
前記複数の論理ミドルボックスは、ファイヤウォール、ネットワークアドレス変換器、負荷バランサのうちの１つである請求項１１に記載のミドルボックス要素。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明はミドルボックスを構成設定するネットワーク制御システムに関する。
【背景技術】
【０００２】
現代の多くの企業は、スイッチ、ハブ、ルータ、ミドルボックス（例えば、ファイヤウォール）、サーバ、ネットワークステーション、そして、様々なアプリケーションとシステムをサポートするタイミングのネットワークに接続されたデバイスを含む大規模で洗練されたネットワークを有している。バーチャルマシンの移行、動的作業負荷、マルチテナント機能、顧客特有のサービス品質と機密保護設定を含むコンピュータネットワーキングがますます精緻化することはネットワーク制御に対するより良いパラダイムを必要としている。ネットワークは伝統的に個々のネットワーク構成要素の下位レベルの構成設定によって管理されてきた。ネットワーク構成設定はしばしば基幹ネットワークに依存している。例えば、アクセス制御リスト（“ＡＣＬ”）エントリをもつユーザアクセスをブロックすることには、ユーザの現在のＩＰアドレスを知ることが必要である。より複雑なタスクは、より広範なネットワークの知識を必要とする。強制的にゲストユーザのポート８０のトラフィックにＨＴＴＰプロキシを否定させるには現在のネットワークトポロジーと各ゲストのロケーションを知ることが必要になる。この処理は、ネットワークのスイッチング要素が多数のユーザにより共有されている場合には、より困難である。
【０００３】
これに応じて、ソフトウェア定義のネットワーキング（ＳＤＮ）と呼ばれる新しいネットワーク制御のパラダイムに対する動きには成長がある。ＳＤＮパラダイムにおいて、ネットワークで１つ以上のサーバで動作するネットワークコントローラは、各ユーザを基本として共有ネットワークスイッチング要素における転送動作を支配する制御ロジックを制御し、維持し、実現する。ネットワーク管理の決定を行うことをはしばしば、ネットワーク状態の知識を必要とする。管理意思決定を容易にするために、ネットワークコントローラは、ネットワーク状態の概観を作成し維持管理し、管理アプリケーションがネットワーク状態の概観にアクセスする際に用いるアプリケーションプログラミングインタフェース

を提供する。

【０００４】

（データセンタや企業向けネットワークを含む）大規模なネットワークを維持管理する主要な目的のいくつかは、スケーラビリティ、モビリティ、マルチテナント機構である。これらの目標の１つを扱うためにとられる多くの方法は、他の目標の少なくとも１つの達成を妨害してしまうという結果を生んでいる。

【発明の概要】

【０００５】

ある実施例では、ユーザが１つ以上のミドルボックスと共に論理データパスセットを含む論理ネットワークを特定することを可能にするネットワーク制御システムを提供する。そのユーザは、（１）論理転送要素（例えば、論理ルータ、論理スイッチ）とネットワーク内のミドルボックスのロケーションとを含むネットワークのトポロジーと、（２）トラフィックをミドルボックスに転送するためのルーティングポリシーと、（３）異なるミドルボックスに対する構成設定とを指定する。ある実施例のネットワーク制御システムは、複数のネットワークコントローラの組を用いて、ネットワークトポロジーを実現するフローエントリとミドルボックスの構成設定との両方を、管理されるスイッチング要素と分散ミドルボックスとが動作するホストマシンとともに、そのホストマシンの外側で動作する集中型ミドルボックス装置とに分配する。

【０００６】

ある実施例では、ネットワークコントローラは階層的に構成される。ユーザはトポロジーと構成設定情報を、論理コントローラ、又は、論理コントローラに複数のレコードからなるセットとしてその情報を受け渡す入力変換コントローラに入力する。論理コントローラは複数の物理コントローラからなる組と通信可能に結合し、各物理コントローラは１つ以上のホスト装置に構成設定データを配信することを担当している。即ち、各ホスト装置はそのホスト装置に対してマスタとして振る舞う特定の物理コントローラに割当てられる。論理コントローラはどのホスト装置が構成設定を受信する必要があるかを識別し、それから、識別されたホスト装置を管理する物理コントローラに適切な情報を受け渡す。レコードを物理コントローラにエクスポートする前に、論理コントローラはフローエントリデータを変換する。いくつかの実施例では、しかしながらミドルボックス構成設定データは変換されない。

【０００７】

物理コントローラはその情報を受信し、そのデータの少なくとも一部に関して付加的な変換を実行し、その変換されたデータをホスト装置（即ち、ホスト装置において管理されるスイッチング要素とミドルボックス）に受け渡す。ある実施例では、論理コントローラのように、物理コントローラが管理されたスイッチに宛てられたフローエントリを変換するが、ミドルボックス構成設定データについては何の変換も実行しない。しかしながら、ある実施例の物理コントローラは、ミドルボックスに対する付加的なデータを生成する。特に、（例えば、デーモンやアプリケーションのような）ホスト装置で動作する分散ミドルボックスアプリケーションの要素は異なるテナントネットワークに関して複数の別々のミドルボックス処理を実行することができるので、物理コントローラはそのミドルボックスに対する特定の構成設定に対してスライシング識別子を割当てる。このスライシング識別子はまた、管理されるスイッチ要素と通信され、その要素はある実施例ではその識別子をミドルボックスに宛てられたパケットに付加する。

【０００８】

先の概要では、簡単な導入部としての役割を果たすことが意図された本発明のいくつかの実施例に焦点を当てた。しかしながら、このことがこの明細書において開示される発明の全ての主題の導入部や概要であることを意味するものではない。続く詳細な説明とその詳細な説明で参照される図面はさらに、その概要で説明した実施例とともに他の実施例に関しても説明するものである。従って、この明細書により説明する全ての実施例を理解するためには、概要と詳細な説明と図面の十分なレビューが必要である。さらに、発明の主

題は、その概要と詳細な説明と図面における例示的に示された詳細な事項より限定されるものではなく、むしろ添付の請求の範囲により規定されるものである。なぜなら、発明の主題は、その主題の趣旨を逸脱することなく他の具体的な形でも実施可能なものであるからである。

【０００９】

本発明の新規な特徴は添付の請求の範囲により説明されるものであるが、説明目的のために、本発明のいくつかの実施例が次の図面において説明される。

【図面の簡単な説明】

【００１０】

【図１】いくつかの実施例の論理ネットワークトポロジーと、ネットワーク制御システムによる構成設定後にこの論理ネットワークを実装する物理ネットワークとを概念的に示す図である。

【図２】ユーザの指定に従って論理ネットワークを実行するために、管理されるスイッチング要素と分散ミドルボックス要素とを（集中型ミドルボックスとともに）構成設定するいくつかの実施例のネットワーク制御システムを概念的に示す図である。

【図３】、

【図４】、

【図５】論理ネットワーク内のミドルボックスに関する情報をネットワーク制御システムにユーザが情報を入力する例とネットワーク制御信号内をデータが通過する変換とを概念的に示す図である。

【図６】いくつかの実施例のネットワークコントローラのアーキテクチュアの例を示す図である。

【図７】、

【図８】、

【図９】いくつかの実施例における、第１の仮想マシンから第２の仮想マシンへとパケットを送信するために実行される異なる動作を概念的に示す図である。

【図１０】本発明のいくつかの実施例が実装される電子システムを概念的に示す図である。

【発明を実施するための形態】

【００１１】

次に示す本発明の詳細な説明において、本発明の種々の詳細や、例や、実施例が説明される。しかしながら、当業者にとって、本発明は説明される実施例によって限定されるものではなく、本発明はここで検討される特定の詳細や例のいくつかがない場合にも実施可能であることは明らかである。

【００１２】

いくつかの実施例では、ユーザが１つ以上のミドルボックス（例えば、ファイヤウォール、負荷バランサ、ネットワークアドレス変換器、不法侵入検出システム、広域ネットワーク（ＷＡＮ）オプティマイザなど）とともに複数の論理データ経路からなる組を含む論理ネットワークを指定可能なネットワーク制御シーケンスを備える。ユーザは、（１）論理転送要素（例えば、論理ルータ、論理スイッチなど）とネットワーク内のミドルボックスロケーションとを含むネットワークトポロジーと、（２）トラフィックをミドルボックスに転送するルーティングポリシーと、（３）異なるミドルボックスに対する構成設定とを指定する。いくつかの実施例のネットワーク制御システムは、ネットワークトポロジーを実現するフローエントリとミドルボックス構成設定との両方を、管理されるスイッチング要素と分散されたミドルボックスとが動作する複数のホストマシンに配信するとともに、前記複数のホストマシンの外部で動作する集中型ミドルボックス機器に対しても同様の配信を行うために、複数のネットワークコントローラからなる組を用いる。

【００１３】

いくつかの実施例では、複数のネットワークコントローラが階層構造となるように構成される。ユーザはトポロジーと構成設定情報とを論理コントローラ、或いは、その情報を

論理コントローラに複数のレコードからなる組として受け渡す入力変換コントローラに入力する。論理コントローラは複数の物理コントローラからなる組と通信可能に結合し、各物理コントローラは１つ以上のホスト装置に構成設定データを配信することを担当している。即ち、各ホスト装置はそのホスト装置に対してマスタとして振る舞う特定の物理コントローラに割当てられる。論理コントローラはどのホスト装置が構成設定を受信する必要があるかを識別し、それから、識別されたホスト装置を管理する物理コントローラに適切な情報を受け渡す。レコードを物理コントローラにエクスポートする前に、論理コントローラはフローエントリデータを変換する。いくつかの実施例では、しかしながらミドルボックス構成設定データは変換されない。

【００１４】

物理コントローラはその情報を受信し、そのデータの少なくとも一部に関して付加的な変換を実行し、その変換されたデータをホスト装置（即ち、ホスト装置において管理されるスイッチング要素とミドルボックス）に受け渡す。ある実施例では、論理コントローラのように、物理コントローラが管理されたスイッチに宛てられたフローエントリを変換するが、ミドルボックス構成設定データについては何の変換も実行しない。しかしながら、ある実施例の物理コントローラは、ミドルボックスに対する付加的なデータを生成する。特に、（例えば、デーモンやアプリケーションのような）ホスト装置で動作する分散ミドルボックスアプリケーションの要素は異なるテナントネットワークに関して複数の別々のミドルボックス処理を実行することができるので、物理コントローラはそのミドルボックスに対する特定の構成設定に対してスライシング識別子を割当てる。このスライシング識別子はまた、管理されるスイッチ要素と通信され、その要素はある実施例ではその識別子をミドルボックスに宛てられたパケットに付加する。

【００１５】

図１はいくつかの実施例の論理ネットワークトポロジー１００と、ネットワーク制御システムによる構成設定後にこの論理ネットワークを実装する物理ネットワークとを概念的に図示している。ネットワークトポロジー１００は説明目的のための単純化されたネットワークである。そのネットワークは、論理Ｌ３ルータ１１５により接続される２つの論理Ｌ２スイッチ１０５、１１０を含む。論理スイッチ１０５は仮想マシン１２０と１２５とを接続する一方、論理スイッチ１１０は仮想マシン１３０と１３５とを接続する。論理ルックアップテーブル１１５はまた、外部ネットワーク１４５に接続される。

【００１６】

加えて、ミドルボックス１４０は論理ルックアップテーブル１１５にアタッチする。当業者であれば、ネットワークトポロジー１００は、ミドルボックスが組み込まれる１つの特定の論理ネットワークトポロジーを表現していることを認識するであろう。種々の実施例では、ミドルボックスは（例えば、）２つの別の要素の間に直接的に配置されても良いし、（例えば、論理ネットワークに入力したりそこから出力される全てのトラフィックを監視し処理するために）外部ネットワークと論理ルータとの間に直接的に配置されても良いし、或いは、より複雑なネットワークにおける別のロケーションに配置されていても良い。

【００１７】

図１に示されるアーキテクチュアにおいて、ミドルボックス１４０は、１つのドメインから別のドメインへの、或いは、外部世界とそのドメインとの間の直接的なトラフィックフロー内に位置してはいない。従って、（例えば、ネットワーク管理者のようなユーザによって）どのパケットが処理のためにミドルボックスに送信されるべきであるのかを決定する論理ルータ１１５に対するルーティングポリシーが特定されないなら、パケットはミドルボックスには送信できないであろう。いくつかの実施例では、ポリシールーティング規則の使用が可能であり、これにより宛先アドレス（例えば、宛先ＩＰ或いはＭＡＣアドレス）をこえてデータに基づくパケットを転送する。例えば、ユーザは、（例えば、ネットワークコントローラアプリケーションプログラミングインタフェース（ＡＰＩ）を介して）論理スイッチ１０５により交換される論理サブセットにおける発信元ＩＰアドレスと

論理スイッチ１０５に接続する論理発信元ポートとを備える全てのパケット、或いは、論理スイッチ１１０により交換される論理サブセットに対して宛てられた外部ネットワーク１４５からのネットワークに入力する全てのパケットは、処理のためにミドルボックス１４０にダイレクトされるべきであることを指定するかもしれない。

【００１８】

論理ネットワークを実現するために、ユーザ（例えば、ネットワーク管理者）により入力される論理ネットワークトポロジーがネットワーク制御システムを介して種々のブロックマシンへと配信される。図１の下段では概念的に論理ネットワーク１００のそのような物理的な実装形１５０を図示している。具体的には、物理的な実装形１５０は、第１のホストマシン１５５、第２のホストマシン１６０、第３のホストマシン１６５を含む幾つかのノードを図示している。３つのノード各々は、論理ネットワーク１００の少なくとも１つの仮想マシンに対してホストとなり、仮想マシン１２０は第１のホストマシン１５５でホストとなり、仮想マシン１２５と１３５とは第２のホストマシン１６０でホストとなり、仮想マシン１３０は第３のホストマシン１６５でホストとなる。

【００１９】

加えて、複数のホストマシン各々は管理されるスイッチング要素（“ＭＳＥ”）を含む。いくつかの実施例では、管理されるスイッチング要素は、１つの以上の論理ネットワークに対する論理転送要素を実現するソフトウェア転送要素である。例えば、ホスト１５５～１６５におけるＭＳＥは、ネットワーク１００の論理転送要素を実装するテーブルを転送することにおいてフローエントリを含む。特に、ホストマシンにおけるＭＳＥは、論理スイッチ１０５と１１０と共に論理ルータ１１５とを実装する。これに対して、いくつかの実施例は、論理スイッチに接続される少なくとも１つの仮想マシンが特定のノードに位置するとき、そのノードにおいて複数の論理スイッチを実装するだけである（即ち、ホスト１５５におけるＭＳＥにおいて論理スイッチ１０５と論理ルータ１１５とを実装するだけである）。

【００２０】

いくつかの実施例の実施形３００はまた、複数のホストマシンに接続するプールノード３４０を含む。いくつかの実施例では、ホストに常駐する複数のＭＳＥは第１のホップ処理を実行する。即ち、これらのＭＳＥは、仮想マシンから送信された後、パケットが到達する第１の転送要素であり、第１のホップにおける論理スイッチングとルーティングとの全てを実行することを試みる。しかしながら、いくつかの場合には、特定のＭＳＥはネットワークに対する論理転送情報の全てを含むフローエントリを格納しないかもしれず、それ故に、特定のパケットをどのように処理するのかを知らないかもしれない。いくつかのそのような実施例では、ＭＳＥはパケットをさらなる処理のためにプールノード３４０に送信する。これらプールノードは内部的に管理されるスイッチング要素であり、それはいくつかの実施例ではソフトウェアのエッジスイッチング要素よりも、論理ネットワークの大きな部分を包含するフローエントリを格納する。

【００２１】

ネットワーク１００の複数の仮想マシンが常駐する複数のホストにまたがる論理スイッチング要素の分散と同様に、ミドルボックス１４０もこれらホスト１５５～１６５の複数のミドルボックス要素にまたがって分散される。いくつかの実施例では、ミドルボックスモジュール（或いは、複数のモジュールの組）は、（例えば、ホストのハイパーバイザで動作する）複数のホストマシンに常駐する。

【００２２】

前述のように、いくつかの実施例のネットワーク制御システムは、複数の分散転送要素（ＭＳＥ）と複数のミドルボックスとを構成設定するのに用いられる。３つのホスト１５５～１６５の各々は、複数のＭＳＥに対するフローエントリと複数のミドルボックスに対する構成設定情報とを受信する特定の物理コントローラに割当てられ、データについての必要な変換を実行し、そのデータをホスト上の要素に受け渡す。

【００２３】

図１は複数のホスト１５５～１６５にまたがって実装されるただ１つの論理ネットワークを図示しているが、いくつかの実施例は複数のホストからなる組にまたがる数多くの論理ネットワーク（例えば、異なるテナントに対して）を実装する。そのようなものとして、特定のホスト上のミドルボックス要素は実際には複数の異なる論理ネットワークに属する複数の異なるミドルボックスに対する構成設定を格納するかもしれない。例えば、ファイヤウォール要素は２つ（以上の）異なるファイヤウォールを実装するために仮想化されても良い。これらは、ミドルボックス要素が（同種の）複数の“仮想”ミドルボックスへとスライスされ分割されるように、２つの分離したミドルボックス処理として効率的に動作するであろう。

【００２４】

加えて、ホスト側でのＭＳＥが複数のパケットをミドルボックスに送信するとき、いくつかの実施例ではそのパケットにスライス識別子（或いはタグ）を付加し（例えば、先頭に付加し）、複数の仮想ミドルボックスのいずれでそのパケットが送信されるのかを識別する。単一の論理ネットワークに対して同じミドルボックス要素に複数のミドルボックス（例えば、２つの異なる負荷バランサ）が実装されるとき、スライス識別子は、そのパケットが属する論理ネットワークよりはむしろ特定のミドルボックススライスを識別する必要がある。異なる実施例では、複数のミドルボックスに対して異なるスライス識別子を用いても良い。

【００２５】

いくつかの実施例では、これらのスライス識別子はネットワーク制御システムにより割当てられる。ミドルボックス１４０のような分散ミドルボックスに対し、ミドルボックス要素は一般にそのホスト上のＭＳＥからのパケットを受信するだけなので、スライス識別子は、特定のミドルボックス要素（とＭＳＥ）を管理する物理コントローラにより、各ミドルボックス要素に対して別々に割当てられる。集中型ミドルボックスの場合（即ち、複数のＭＳＥが複数のパケットを送信する別々の物理機器）、単一のスライス識別子がその機器で動作する仮想ミドルボックスに対して用いられるであろう。いくつかの実施例では、その機器を管理する物理コントローラはこの識別子を割当てて、それからこれをネットワーク制御システムを介して他の物理コントローラに配信し、他の物理コントローラがこの情報を複数のＭＳＥに受け渡すようにする。

【００２６】

上述のことはいくつかの実施例のネットワークにおける論理ミドルボックスの実施形の例を例示したものである。以下、いくつかのより詳細な実施例について説明する。セクションＩではファイヤウォールを含む論理ネットワークを実装するためにネットワークを構成設定するいくつかの実施例のネットワーク制御システムについて説明する。セクションＩＩではいくつかの実施例のネットワークコントローラのアーキテクチュアについて説明する。次に、セクションＩＩＩではパケットがミドルボックスを通過するときに２つの仮想マシンの間でのパケット処理について説明する。最後にセクションＩＶでは本発明のいくつかの実施例が実装される電子システムについて説明する。

【００２７】

Ｉ．ネットワーク制御システム

上述のように、幾つかの実施例はミドルボックスと管理されるネットワークのための管理されるスイッチング要素のプロビジョニングを行うためにネットワーク制御システムを用いる。幾つかの実施例では、ネットワーク制御システムは複数のコントローラからなる階層的な組であり、その階層構造の各レベルでは管理されるスイッチング要素とミドルボックスのプロビジョニングにおいて異なる機能を実行する。

【００２８】

図２はユーザの指定に従って論理ネットワークを実現するために、管理されるスイッチング要素と分散ミドルボックス要素とを（集中型ミドルボックスとともに）構成設定するいくつかの実施例のネットワーク制御システムを概念的に示す図である。図示のように、ネットワーク制御システムは入力変換コントローラ２０５、ミドルボックス構成設定イン

タフェース２０７、論理コントローラ２１０、物理コントローラ２１５、２２０、ホスト２２５～２３５を含む。図示のように、ホスト２２５～２３５は、複数の管理されるスイッチング要素と複数の分散ミドルボックス要素との両方を含む。幾つかの実施例では、ネットワーク制御システムは、それぞれが単一の物理コントローラに結合される複数の集中型ミドルボックス（例えば、物理的な機器、個々の仮想マシンなど）を含んでも良い。

【００２９】

幾つかの実施例では、ミドルボックス構成設定インタフェース２０７は実際には、入力変換コントローラ２０５の一部である。この図では、２つが別々に図示されており、それらは異なる入力を受信し、ユーザと通信を行うための異なるＡＰＩを含んでいる。幾つかの実施例では、ネットワーク制御システムにおける複数のコントローラ各々は、入力変換コントローラと論理コントローラと物理コントローラとの内の少なくともいずれかとして機能する能力をもっている。即ち、各コントローラマシンは、異なるコントローラタイプのいずれかの機能を実装する必要なアプリケーションスタックを含むが、いつの時にもそれらアプリケーションスタックの１つだけが用いられる。また、幾つかの実施例では、所与のコントローラは複数タイプのコントローラの特定の１つのものとして（例えば、物理コントローラとして）動作する機能をもつだけでも良い。加えて、複数のコントローラの異なる組み合わせが同じ物理マシンで実行されても良い。例えば、入力変換コントローラ２０５、ミドルボックス構成設定インタフェース２０７、論理コントローラ２１０はユーザがインタラクトする同じコンピュータ機器で実行されても良い。

【００３０】

さらに、図２（と後続の図３～５）で図示された複数のコントローラ各々は単一のコントローラとして図示されているが、これらコントローラ各々は実際には分散構成で動作し、論理コントローラ、物理コントローラ、或いは、入力変換コントローラの処理を実行するコントローラクラスタであっても良い。

【００３１】

幾つかの実施例の入力変換コントローラ２０５は、ユーザから受信したネットワーク構成設定情報を変換する入力変換アプリケーションを含む。例えば、ユーザは図１に示されているような、どのマシンがどの論理ドメインに属しているのかに関する仕様を含むネットワークトポロジーを指定するかもしれない。これにより効果的に論理データパスセット、或いは、複数の論理転送要素からなる組を指定する。複数の論理転送要素各々に関し、ユーザは論理スイッチに接続する複数のマシンや他の要素を指定する（即ち、複数の論理ポートがその論理スイッチに対して割当てられる）。幾つかの実施例では、ユーザはまた複数のマシンのためのＩＰアドレスも指定する。

【００３２】

例えば、ユーザは図１に示されているようなネットワークトポロジーを入力するかもしれず、そのトポロジーでは、複数のマシンが複数の論理スイッチに接続され、論理ルータが２つの論理スイッチに接続し、１つ以上のミドルボックスが同様に論理ルータの複数のポートに接続される。フロー生成の列で図示されるように、入力変換コントローラ２０５は入力されたネットワークトポロジーを、そのネットワークトポロジーを記述する論理制御プレーンへと変換する。幾つかの実施例では、論理制御プレーンデータはデータベーステーブルレコードのセットとして（例えば、ｎＬｏｇ言語において）表現される。特定の仮想マシンのネットワークへのアタッチを記述する制御プレーンにおける入力は、特定のＭＡＣアドレスＢが特定の論理スイッチの特定の論理ポートＸに位置していることを伝えている。

【００３３】

ミドルボックス構成設定インタフェース２０７は、ユーザからミドルボックス構成設定入力を受信する。幾つかの実施例では、異なるミドルボックス各々（例えば、異なるプロバイダからのミドルボックス、異なるタイプのミドルボックス）はミドルボックス実装に特有の異なるＡＰＩを有しているかもしれない。即ち、異なるミドルボックスの実施形はユーザに呈示される異なるインタフェースをもつ（即ち、ユーザは異なる特定のミドルボ

ックスに対して異なるフォーマットで情報を入力しなければならないであろう）。図２のミドルボックスデータ生成の例で図示されるように、ユーザは、ミドルボックスＡＰＩによりミドルボックス構成設定データへと変換されるミドルボックス構成設定を入力する。

【００３４】

幾つかの実施例では、構成設定インタフェース２０７により変換されるようなミドルボックス構成設定データはまた、複数のレコードからなる組であり、各レコードは特定の規則を指定する。幾つかの実施例では、これらのレコードは、管理されるスイッチング要素へと伝搬されるフローエントリに類似のフォーマットをもっている。事実、幾つかの実施例は複数のコントローラで同じアプリケーションを用いて、フローエントリ関してはファイヤウォール構成設定レコードを、そして、そのレコードに関しては同じテーブルマッピング言語（例えば、ｎＬｏｇ）を伝搬する。

【００３５】

この図は論理コントローラに送信されるミドルボックス構成設定データを図示しているが、幾つかの実施例の幾つかの集中型ミドルボックスはミドルボックスデバイスとの直接的なインタフェースを介してのみアクセス可能である。即ち、論理コントローラに送信され、ネットワーク制御システムを介して配信される構成設定を入力するというよりはむしろ、ユーザはミドルボックスデバイスに直接に構成設定を入力する。そのような場合、ユーザは依然としてルーティングポリシーを入力してパケットをミドルボックスにネットワークトポロジー構成設定の一部として送信することが必要であろう。そのような幾つかの実施例では、ネットワーク制御システムは依然として、以下に説明するように、ミドルボックスに対するスライシングデータ（即ち、仮想化識別子）を生成するであろう。これに対して、幾つかの実施例では、ユーザはミドルボックスに対するスライシングデータを構成設定し、そして、ミドルボックス或いはユーザがこの情報をネットワーク制御システムに提供する。

【００３６】

幾つかの実施例では、各論理ネットワークは特定の論理コントローラ（例えば、論理コントローラ２１０）により支配される。管理されるスイッチング要素に対してフロー生成に関して、幾つかの実施例の論理コントローラ２１０は入力変換コントローラ２０５から受信される論理制御プレーンを論理転送プレーンデータに変換し、論理転送プレーンデータをユニバーサル制御プレーンデータに変換する。幾つかの実施例では、論理コントローラアプリケーションスタックは、第１の変換を実行する制御アプリケーションと、第２の変換を実行する仮想化アプリケーションとを含む。幾つかの実施例では、これらのアプリケーションの両方は、複数のテーブルからなる第１の組を複数のテーブルからなる第２の組にマッピングする規則エンジンを用いる。即ち、異なるデータプレーンは複数のテーブル（例えば、ｎＬｏｇテーブル）として表現され、コントローラアプリケーションは複数のプレーン間で変換を行うためにテーブルマッピングエンジンを用いる。幾つかの実施例では、制御アプリケーションと仮想化アプリケーションとの両方は同じ規則を用いて変換を実行する。

【００３７】

幾つかの実施例では、論理転送プレーンデータは、論理レベルで記述される複数のフローエントリからなる。論理ポートＸでのＭＡＣアドレスＢに関して、論理転送プレーンデータは、パケットの宛先はＭＡＣ　Ｂに一致しているなら、パケットをポートＸに転送することを特定するフローエントリを含むかもしれない。

【００３８】

論理転送プレーンから物理制御プレーンへの変換は、幾つかの実施例では、レイヤをフローエントリに付加する。そのフローエントリにより、そのフローエントリでプロビジョニングされる管理されるスイッチング要素が物理レイヤポート（例えば、仮想インタフェース）で受信されるパケットを論理ドメインへと変換し、この論理ドメインにおける転送を実行することを可能にする。即ち、トラフィックパケットは物理レイヤでネットワーク内で送受信される一方、その転送決定はユーザにより入力される論理ネットワークトポロ

ジーに従ってなされる。論理転送プレーンから物理制御プレーンへの変換により、幾つかの実施例では、ネットワークのこの側面を可能にしている。

【００３９】

図示のように、論理コントローラは論理転送プレーンデータをユニバーサル物理制御プレーンに変換する一方、物理コントローラはユニバーサル物理制御プレーンデータをカスタマイズされた物理制御プレーンに変換する。幾つかの実施例のユニバーサル物理制御プレーンデータは、幾つかの実施例の制御システムが大多数の（例えば、数千の）管理されるスイッチング要素を含むときでさえ、スケーリングして論理データパスセットを実装することを可能にするデータプレーンである。管理されるスイッチング要素における相違と管理されるスイッチング要素のロケーションの詳細との内の少なくともいずれかを考慮せずに物理制御プレーンデータを表現するために、ユニバーサル物理制御プレーンは異なる管理されるスイッチング要素の共通の特徴を概念化する。

【００４０】

先に注記した例（ＭＡＣ　Ｂの論理ポートＸへのアタッチメント）に関して、ユニバーサル物理制御プレーンは幾つかのフローエントリに関与する。第１のエントリは、もしパケットが（例えば、特定の論理発信元ポートで受信されるパケットに基づいて）特定の論理データパスセットに一致し、その宛先アドレスがＭＡＣ　Ｂに一致するなら、そのパケットを論理ポートＸに転送することを述べている。これは論理データパスセット（物理ポートから論理ポートへの変換）によりその一致を、論理ドメインにおけるその解析を実行する転送エントリへ付加する。このフローエントリは、いくつかの実施例では、ユニバーサルのそしてカスタマイズされた物理制御プレーンにおけるのと同じであろう。

【００４１】

付加的なフローが、物理発信元ポート（例えば、ホストマシンの仮想インタフェース）を（ＭＡＣ　Ａから受信するパケットに関して）論理発信元ポートに一致させるとともに、論理ポートＸを（ＭＡＣ　Ａに送信されるパケットに関し）管理される物理スイッチの特定の宛先ポイントに一致させるために生成される。しかしながら、これら物理発信元ポートと宛先ポイントとは、管理されるスイッチング要素を含むホストマシンに固有である。そのようなものとして、ユニバーサル物理制御プレーンエントリは、論理発信元ポートに対する概念上の物理ポート（即ち、特定の物理ホストマシンのいずれにも固有ではないポートの一般的な抽象概念）と共に、論理宛先ポイントの一般的物理宛先ポートへのマッピングをも含む。

【００４２】

これに対して、ミドルボックス構成設定データは幾つかの実施例では論理コントローラにより変換されず、一方、他の実施例では、論理コントローラはミドルボックス構成設定データレコードの少なくとも最低限の変換を実行する。多くのミドルボックスでのパケット処理、変換、解析規則は、パケットのＩＰアドレス（或いはＴＣＰ接続状態）で動作し、ミドルボックスに送信されるパケットはこの情報を露出させる（即ち、論理ポート情報内にカプセル化しない）ので、ミドルボックスの構成設定は論理データプレーンから物理データプレーンへの変換を必要としない。従って、同じミドルボックス構成設定データがミドルボックス構成設定インタフェース２０７から論理コントローラ２１０に、それから物理コントローラ２１５、２２０へと受け渡される。

【００４３】

物理制御プレーンデータをミドルボックス構成設定データとともに配信するために、論理コントローラはどのホストマシン（と従ってどの物理コントローラ）がどのフローエントリとどのミドルボックス構成設定情報とを受信する必要があるのかを識別しなければならない。いくつかの実施例では、論理コントローラ２１０は論理ネットワークの記述とその物理ネットワークの物理的実装の記述とを格納している。論理コントローラは分散ミドルボックスに関する１つのミドルボックス構成設定レコードを受信し、種々のノードのいずれかがその構成設定情報を受信する必要があるのかを識別する。

【００４４】

いくつかの実施例では、ミドルボックス構成設定全体が複数のホストマシンの全てにおけるミドルボックス要素に分配されるので、少なくとも１つの仮想マシンが常駐し、パケットがファイヤウォールの利用を必要とするマシンの全てを識別する。一般に、識別されたマシンは、（例えば、図１に示したミドルボックスに関し）ネットワークにおける仮想マシンの全てに対するホストである。しかしながら、いくつかの実施例は、もし、そのネットワークトポロジーがミドルボックスがあるホストマシンでは決して必要とはされないようなものであれば、そのネットワークで仮想マシンのサブセットを識別するかもしれない。いくつかの実施例は、どのホストマシンがレコード毎を基本として構成設定データを送信するのかについての決定を行う。即ち、特定の規則各々は仮想マシンのサブセットにのみ（例えば、特定の仮想マシン又は複数の仮想マシンのサブセットから発信するパケットにのみ）適用し、これらの仮想マシンを実行させるホストだけがそのレコードを受信する必要がある。

【００４５】

同様に、論理コントローラは物理制御プレーンにおいてどのノードが各フローエントリを受信すべきなのかを識別する。例えば、論理スイッチ１０５を実装するフローエントリはホスト１５５、１６０に分配されるが、図１におけるホスト１６５には分配されない。

【００４６】

一度、論理コントローラが特定のノードを識別しレコードを受信するなら、その論理コントローラはこれら特定のノードを管理する特定の物理コントローラを識別する。幾つかの実施例では、各ホストマシンは割当てられたマスタ物理コントローラをもつ。従って、もし論理コントローラが第１と第２のホストのみを構成設定データの宛先として識別するなら、これらのホストに対する物理コントローラが識別されて論理コントローラからのデータを受信するであろう（、そして、他の物理コントローラはこのデータを受信しないであろう）。集中型ミドルボックスに関し、論理コントローラは、そのミドルボックスを実装する機器を管理する（単一の）物理コントローラを識別するだけが必要となる。集中型ミドルボックスがクラスターとして（例えば、複数のリソースのセット、マスターバックアップクラスターなど）実現されるとき、そのクラスターの各ミドルボックス機器は構成設定データを受信する。そのクラスターのミドルボックスは全て、幾つかの実施例では単一の物理コントローラにより管理されるが、別の実施例では、異なる物理コントローラがクラスター内の異なるミドルボックスを管理する。

【００４７】

ミドルボックス構成設定データを複数のホストに供給するために、幾つかの実施例の論理コントローラはデータを（その論理コントローラにおけるテーブルマッピングエンジンの出力にアクセスするエクスポートノードを用いて）物理コントローラへとプッシュする。他の実施例では、複数の物理コントローラが、その論理コントローラのエクスポートモジュールからの構成設定データを（例えば、その構成設定データが利用可能であることを示す信号に応答して）要求する。

【００４８】

前述のように、物理コントローラ２１５、２２０の各々が（例えば、ホストマシン内に位置する）１つ以上の管理されるスイッチング要素のマスタである。この例では、第１の物理コントローラ２１５は、ホストマシン２２５、２３０において管理されるスイッチング要素のマスタであり、一方、第２の物理コントローラ２２０は、ホストマシン２３５において管理されるスイッチング要素のマスタである。幾つかの実施例では、物理コントローラは論理ネットワークに関するユニバーサル物理制御プレーンデータを受信し、このデータを、（物理コントローラが特定の論理ネットワークに関するデータを受信しない付加的な管理されるスイッチング要素を管理することもあるので、）その物理コントローラが管理し、そのデータを受信する必要のある特定の管理されるスイッチに対するカスタマイズされた物理制御プレーンデータへと変換する。他の実施例では、物理コントローラは適切なユニバーサル物理制御プレーンデータを、（例えば、ホストマシンで実行するシャーシコネクタの形式で）変換それ自身を実行する能力を含む、管理されるスイッチング要素

へと受け渡す。

【００４９】

ユニバーサル物理制御プレーンデータをカスタマイズされた物理制御プレーンへと変換することには、フローエントリにおける種々のディスプレイのカスタマイズが関与する。ユニバーサル物理制御プレーンエントリは、そのエントリが異なるスイッチング要素に対しては異なるいずれかのデータに対する一般化した抽象概念を含むので、管理されるスイッチング要素のいずれにも適用可能であるが、カスタマイズされた物理制御プレーンエントリはそのエントリが送信される特定の管理されるスイッチング要素に固有の代用データを含む。例えば、物理コントローラは、ユニバーサル物理制御プレーン発信元及び宛先ポート統合エントリにおける物理層ポイントをカスタマイズして、具体的なホストマシンの実際の物理層ポート（例えば、仮想インタフェース）を含む。

【００５０】

図２に示されるように、物理コントローラ２１５、２２０は、割当てられたホストマシンにおいて情報を管理されるスイッチング要素とミドルボックスの両方に受け渡す。幾つかの実施例では、ミドルボックス構成設定と物理制御プレーンデータとはホストマシンで駆動する同じデータベースに送信され、管理されるスイッチング要素とミドルボックスモジュールとはそのデータベースから適切な情報を取り出す。同様に、集中型ミドルボックスに関して、物理コントローラはミドルボックス構成設定データをミドルボックス機器（例えば、構成設定データを格納するミドルボックスにおけるデータベース）へと受け渡す。

【００５１】

管理されるスイッチング要素に受け渡されたカスタマイズされた物理制御プレーンデータは、幾つかの実施例では、その管理されたスイッチング要素がパケットをミドルボックスに送信するのを可能にするためのアタッチメントとスライシング情報とを含む。このスライシングデータは、図２のミドルボックススライス情報生成の列に示されているように、物理コントローラ内で生成され、また、幾つかの実施例では物理コントローラとともにミドルボックスへも送信される。ミドルボックス構成設定は分散ミドルボックス要素内でミドルボックスインスタンスを仮想化するために用いられるので、そのミドルボックス要素は（例えば、異なるテナントネットワークのため、単一テナントネットワーク内の異なる論理ミドルボックスのため）複数の別々なミドルボックス処理を一度に実行させるかもしれない。

【００５２】

本質的に、スライシング情報は、管理されるスイッチング要素がミドルボックスに送信するパラメータに付加するためのタグである。このタグはミドルボックスにより実行される（潜在的には）幾つかの処理のいずれに送信されるべきであるのかを示す。従って、ミドルボックスがパケットを受信するとき、そのタグにより、ミドルボックスが、そのパケットでの操作を実行するために、パケット処理、解析、変更などの規則の適切なセットを用いることができるようになる。いくつかの実施例は、スライシング情報をパケットに付加するというよりはむしろ、各ミドルボックスインスタンスのための管理されるスイッチング要素の異なるポートを定義して、（分散構成の場合には）本質的にはファイヤウォールに対して宛てられたトラフィックをスライスするために複数のポートを用いるか、或いは、（集中構成の場合には）集中型機器の異なるポートに接続して複数のインスタンスの間での区別をつける。

【００５３】

スライシングデータを管理されるスイッチング要素にカスタマイズされた物理制御プレーンデータの一部として送信するために、幾つかの実施例では、物理コントローラはスライシングを特定するフローエントリをミドルボックスに付加する。具体的には特定の管理されるスイッチング要素に関し、フローエントリは、（例えば、ＶＬＡＮタグや類似のタグでも良いのであるが）特定のミドルボックスに関するスライシングタグを、ミドルボックスに接続するポートの一致に基づいて特定のミドルボックスにパケットを送信する前に

そのパケットに付加することを指定すると良い。

【００５４】

幾つかの実施例では、アタッチメント情報は、管理されるスイッチング要素がパケットをミドルボックスに送信可能にするフローエントリを含む。分散ミドルボックスの場合には、管理されるスイッチング要素と同じ物理マシンにミドルボックスがあると、そのミドルボックスと管理されるスイッチング要素とは、幾つかの実施例では、転送されるパケットを介してソフトウェアポートの抽象概念をネゴシエートする。幾つかの実施例では、管理されるスイッチング要素（或いはミドルボックス要素）はこの情報を物理コントローラにまで受け渡し、その物理コントローラが（カスタマイズされた物理制御プレーンエントリに関する特定のソフトウェアポートを用いて）カスタマイズされた物理制御プレーンデータにおける情報を使用可能にする。

【００５５】

集中型ミドルボックスに関して、幾つかの実施例では管理されるスイッチング要素とミドルボックスの両方にトンネルアタッチメントデータを提供する。幾つかの実施例では、ミドルボックスは、種々のホストマシンがパケットをミドルボックスに送信するのに用いるであろうトンネルカプセル化のタイプを知る必要がある。幾つかの実施例では、ミドルボックスは受け入れられるトンネルプロトコル（例えば、ＳＴＴ、ＧＲＥなど）のリストをもち、選択されるプロトコルが管理されるスイッチング要素とミドルボックスとの間で調整される。そのトンネルプロトコルは、ミドルボックス構成設定の一部としてユーザにより入力されても良いし、また、別の実施例ではネットワーク制御システムにより自動的に決定されても良い。物理コントローラはまた、管理されるスイッチング要素がミドルボックスに送信するために適切にパケットをカプセル化するために、トンネルカプセル化情報をカスタマイズされた物理制御プレーンフローエントリに付加するであろう。

【００５６】

物理コントローラからカスタマイズされた物理制御プレーンデータを受信する際、管理されたスイッチング要素はカスタマイズされた物理制御プレーンデータの物理転送プレーンデータへの変換を実行する。幾つかの実施例では、物理転送プレーンデータは、スイッチング要素（物理ルータ又はスイッチ、又は、ソフトウェアスイッチング要素）とそのスイッチング要素が実際に受信パケットと一致し、その一致に基いてそのパケットについての動作を実行することを示す転送テーブル内に格納されるフローエントリである。

【００５７】

ミドルボックスはその構成設定データを物理コントローラから受信し、幾つかの実施例ではその構成設定データを変換する。ミドルボックス構成設定データは特定の言語ではパケット処理、解析、変形などの規則を表現するために、ミドルボックスの制御プレーンＡＰＩを介して受信される。幾つかの実施例の（分散型と集中型との内の少なくともいずれかの）ミドルボックスはこれらの規則をより最適化されたパケット分類規則へとコンパイルする。幾つかの実施例では、この変換は、物理制御プレーンから物理転送プレーンデータへの変換に類似している。パケットがミドルボックスにより受信されるとき、効率的かつ迅速にそのパケットにおける動作を実行するためにコンパイルされた最適化された規則を適用する。

【００５８】

図２に示されているように、ミドルボックスはまたスライシング情報を内部スライスバインディングへと変換する。幾つかの実施例では、ミドルボックスはミドルボックス内の状態（例えば、アクティブＴＣＰ接続、種々のＩＰアドレスについての統計など）を識別するために（パケットの先頭に添付されるタグとは異なる）自分自身の内部識別子を用いる。命令を受信し新しいミドルボックスインスタンスと、その新しいインスタンスに対して（パケットで用いられる）外部識別子とを作成する際に、幾つかの実施例は自動的に新しいミドルボックスインスタンスを作成し、そのインスタンスに内部識別子を割当てる。加えて、そのミドルボックスは、外部スライス識別子を内部スライス識別子にマッピングするインスタンスに対するバインディングを格納する。

【００５９】
図３～図５は論理ネットワーク内のミドルボックスに関する情報をネットワーク制御システムにユーザを入力する例とネットワーク制御信号内をデータが通過する変換とを概念的に示す図である。特に、図３はユーザが論理ネットワークトポロジー３０５とルーティングポリシー３１０とをネットワーク制御システムに入力する例を例示している。論理ネットワークトポロジー３０５は、図１に示すトポロジー、つまり、論理ルータＣにより接続された２つの論理スイッチＡ、Ｂと、そのルータにぶら下がるミドルボックスＤを備えたものに類似している。その論理トポロジーに示されるように、ミドルボックスＤはポートＫで論理ルータにアタッチする。
【００６０】
論理ルータがどのパケットをミドルボックスに送信すべきであるのかを示すために、ルーティングポリシー３１０がユーザにより入力される。そのミドルボックスが２つの論理転送要素の間（即ち、論理ルータと論理スイッチとの間）の論理ワイヤに位置するなら、その論理ワイヤにより送信される全てのパケットは自動的にミドルボックスに転送されるであろう。しかしながら、ネットワークトポロジー３０５におけるもののような帯域外のミドルボックスに関しては、論理ルータは、特定のポリシーがユーザにより指定されたときに、そのミドルボックスへとパケットを送信するだけであろう。
【００６１】
ルータとスイッチとは通常、パケットの宛先アドレス（例えば、ＭＡＣアドレス或いはＩＰアドレス）に従ってパケットを転送する一方、ポリシールーティングにより、そのパケットに格納された他の情報（例えば、発信元アドレス、発信元アドレスと宛先アドレスとの組み合わせなど）に基いて転送決定がなされることが可能になる。例えば、ユーザは、特定のサブネットワークにおける発信元ＩＰアドレスもつ全てのパケット、又は、特定のサブネットワークのセットには一致しない宛先ＩＰアドレスをもつ全てのパケットが、ミドルボックスに転送されるべきであることを指定するかもしれない。この具体的な場合では、ユーザにより指定されるルーティングポリシー３１０は、サブネットワークＡにおける発信元ＩＰをもつトラフィックと論理ポートＬの発信元コンテクスト（即ち、論理スイッチＡから到来する）とをミドルボックスにルーティングする。発信元ＩＰアドレスはパケットをミドルボックスにルーティングするのに十分であるが、発信元ポートはミドルボックスから戻ってくるパケットがミドルボックスに再送信される（即ち、決して終わらないループ）のを防止する。ミドルボックスからのパケットは異なる発信元ポートをもち、それ故に、ポリシー３１０によりルーティングされないであろう。
【００６２】
図示されるように、ルーティングポリシーは論理制御プレーンデータ３１５として、論理コントローラ３２０に（例えば、入力変換コントローラから）送信される。この例において、制御プレーンエントリは“発信元ポートＬで受信したサブネットワークＡにおける発信元ＩＰアドレスをもつパケットをポートＫにおけるミドルボックスＤに送信する”ことを述べる。このＬ３（論理ルーティング）制御プレーンエントリは、ルーティングポリシー３１０を、ポートＫに位置したミドルボックスのネットワークトポロジーに組み合わせる。図示のように、論理コントローラ３２０は、論理制御プレーンエントリ３１５を論理転送プレーンエントリ３２５に第１の変換を行う。説明したように、幾つかの実施例では、論理コントローラ３２０におけるテーブルマッピング規則エンジンはこの変換を実行する。即ち、エントリ３１５は第１のデータベーステーブルレコードであり、そのレコードは規則エンジンを介してエントリ３２５にマッピングされる。論理転送プレーンエントリ３２５は、一致→動作のフォーマットにおけるフローエントリであり、“発信元ＩＰが｛Ａ｝に一致し発信元がポートＬに一致するなら→ポートＫに転送”することを記述している。ネットワークは論理プレーンにおける転送を実行するので、フローエントリはパケットを論理ルータの論理ポートに送信する。
【００６３】
次に、論理コントローラ３２０は論理転送プレーンエントリ３２０を複数のユニバーサ

ル制御プレーンエントリ３３０～３４０からなる組へと変換する。第１のエントリ３３０はユニバーサル物理制御プレーンにおける転送エントリであり、“一致するＬ３　Ｃと発信元ＩＰが｛Ａ｝に一致し発信元がポートＬに一致するなら→ポートＫに転送”することを記述している。このエントリは、Ｌ３ルータによる一致に転送エントリを付加し、フローエントリにより動作するパケットが異なる論理ネットワークの一部ではないことを保証する。

【００６４】

加えて、論理コントローラは発信元と宛先ポートの統合エントリ３３５、３４０を付加する。これらのエントリはユニバーサル物理制御プレーンの一部であるので、これらのエントリにおける物理ポート情報は一般的である。これらのエントリは、ホストマシンにおけるミドルボックスに接続されるソフトウェアポートを介して受信されるパケットを論理発信元ポートＫにマッピングする発信元ポート統合エントリ３３５を含む。同様に、宛先ポート統合エントリ３４０は、ポートＫに転送されるパケットをミドルボックスに接続されるソフトウェアポートにマッピングする。このポートは異なるホストマシンにおける異なる具体的な識別子をもつかもしれないので、ユニバーサル制御プレーンレベルにおいて、そのようなポートの汎用抽象概念を用いて表現される。ユニバーサル物理制御プレーンは、ＶＭ１が配置される（一般的な）仮想インタフェースから受信するパケットを、Ｌ２　Ａ論理スイッチにおける発信元ポートにマッピングするための発信元マッピングや、Ｌ２　Ａ論理スイッチにはない宛先をもつパケットを論理的にはＬ３　Ｃ論理ルータに接続される宛先ポートにマッピングするＬ２転送エントリのような他の種々のエントリを含む。

【００６５】

論理制御プレーンから論理転送プレーンへの変換と同様に、論理コントローラ３２０はテーブルマッピング規則エンジンを用いた第２の変換を実行する。幾つかの実施例では、第１の変換は、論理コントローラ内の制御アプリケーションにより実行される一方、第２の変換は仮想化アプリケーションにより実行される。そのような幾つかの実施例では、これら２つのアプリケーションは同じ規則エンジンを用いる。

【００６６】

次に、論理コントローラ３２０は、ネットワーク制御システムにおけるどの物理コントローラがユニバーサル物理制御プレーンのエントリを受信すべきなのかを識別する。例えば、特定の仮想マシンについてＬ２レベルでの発信元と宛先ポートの統合エントリは、特定の仮想マシンに対してホストとなるノードのマスタにある物理コントローラに送信される必要があるだけである。幾つかの実施例では、Ｌ３ルータに対するエントリは複数のマシンの全てに送信され、それ故に、論理コントローラ３２０は、論理ネットワークについてのユニバーサル物理制御プレーンデータを受信してエントリ３３０～３４０を受信する複数の物理コントローラの全てを識別する。これに対して、幾つかの場合では、ミドルボックスは（例えば、そのミドルボックスが複数のノードにおけるトラフィックを処理する必要があるだけであるために）これらのノードのサブセットに実装されるだけであるかもしれず、それ故に、これらのエントリはそのサブセットにおけるノードを管理する物理コントローラにエクスポートされるだけである。

【００６７】

エントリ３３０～３４０を受信する際、物理コントローラ３４５（これらのエントリを受信するための複数の物理コントローラの１つ）は、ユニバーサル物理制御プレーンからカスタマイズされる物理制御プレーンへの変換を実行する。図示のように、エントリ３３０は、仕様を要求するフローエントリにおける情報がないので、同じ状態に留まる。これに対して、発信元と宛先ポートの統合エントリ３３５と３４０における一般的なポート抽象概念は、ミドルボックスがエントリ３５５と３６０のために接続するＭＳＥ３５０の特定のソフトウェアポートへと変換される。

【００６８】

図示のように、幾つかの実施例では、ミドルボックスとのソフトウェアポート接続をネ

ゴシエートした後、ＭＳＥは物理コントローラ３４５にまでミドルボックスアタッチメントポート情報３６５を受け渡す。それから、物理コントローラはこの情報を、ユニバーサル物理制御プレーンのレコードをカスタマイズされた物理制御プレーンのレコードへと変換するテーブルマッピングエンジンに対する入力として用いる。物理コントローラはこの情報をＭＳＥに受け渡し、そのＭＳＥはカスタマイズされた物理制御プレーンのエントリをその転送テーブル３７０における物理転送プレーンエントリに変換する。これらの転送テーブルはＭＳＥにより用いられ、受信パケットに一致させ、そのパケットの（例えば、転送、カプセル化など）動作を実行する。ここで図示されてはいないが、付加的なフローエントリは物理コントローラにより生成され、パケットをソフトウェアポートによりミドルボックスに送信する前にスライシングタグをそのパケットに付加する。

【００６９】

図４は、論理ネットワークのミドルボックスＤを構成設定する図３におけるのと同じユーザと、構成設定データのミドルボックスまでの伝搬とを概念的に図示している。図示のように、同じ論理ネットワークトポロジー３０５がミドルボックス規則４０５とともに入力される。ミドルボックス規則４０５は、ミドルボックスＤに関する実施形とミドルボックスのタイプに固有のミドルボックス構成設定インタフェースを介して入力され、ミドルボックスがどのようにパケットを処理するのかについての規則を指定する。例えば、もしミドルボックスがファイヤウォールであるなら、その規則はブロックしたり或いは許可を与えたりする特定の発信元ＩＰを規定する。もし、そのミドルボックスが発信元ネットワークアドレス変換器であるなら、その規則は特定の仮想ＩＰを隠すために現実のＩＰの組を規定する。もし、そのミドルボックスが負荷バランサであるなら、その規則は特定の仮想ＩＰの背後にある特定の組の複数サーバの負荷バランスをとるために用いるための特定のスケジュールリングアルゴリズムを規定するなどである。

【００７０】

図示のように、論理コントローラ３２０は規則４１５を（例えば、データベーステーブルレコードとして）受信する。その論理コントローラは規則４１５を受信すべき特定のノードを識別する。幾つかの実施例では、これはミドルボックスＤを実行するノード全てであり、そのミドルボックスはネットワークの論理転送要素を実装するノードの全てであるかもしれない。しかしながら、幾つかの実施例では、論理コントローラはミドルボックスに対するルーティングポリシーを分析して、（即ち、論理ネットワークにおけるその配置と機能とに基いて）論理転送要素を実行する複数のノードのサブセットだけがミドルボックスを実装する必要があることを判断する。識別されたノードに基いて、論理コントローラは、規則４１５を分配する複数の物理コントローラの組を識別する。

【００７１】

この組の物理コントローラは例示された物理コントローラ３４５を含み、そのコントローラに規則４１５をエクスポートする。論理コントローラ３２０におけるように、物理コントローラ３４５は規則４１５での変換（或いは、少なくとも最低限の変換だけ）を実行しない。しかしながら、新しい構成設定がホストマシンで実行するミドルボックスアプリケーション（例えば、ミドルボックスデーモン）で新しい仮想ミドルボックスの開始を引き起こすので、物理コントローラはミドルボックスインスタンスに対してのスライシング識別子４２０を割当てる。

【００７２】

物理コントローラ３４５は規則４１５とスライシング識別子４２０をＭＳＥ３５０と同じホストマシンのミドルボックス４２５に分配する。ミドルボックスは新しいミドルボックスインスタンスを生成し、規則４１５を（物理コントローラ３４５から受信した他の構成設定規則とともに）ミドルボックスインスタンスに関するデータプレーン規則４３０のコンパイルされたセットへと変換する。幾つかの実施例では、これらデータプレーンの規則４３０はミドルボックスにおけるパケット処理のためのハードコード化された転送テーブルのセットとして効果的に作用する。加えて、ミドルボックス４２５はエントリをそのスライスバインディングテーブル４３５へ負荷する。各インスタンスに関し、ミドルボッ

クス４２５はそれ自身の内部識別子４３５を作成し、物理コントローラにより割当てられたパックされたスライシングＩＤに対するこの内部ＩＤのバインディングをスライスバインディングテーブル４３５に格納する。さらにその上、図示のように、ミドルボックスはホストマシンにおけるＭＳＥ３５０と契約を結び、２つのモジュール間でパケットを転送するためのソフトウェアポートを作成する。

【００７３】

図５は、第２の論理ネットワークのミドルボックスＨを構成設定する第２のユーザと、ミドルボックス４２５までの構成設定データの伝搬とを、概念的に図示している。図示のように、第２のユーザは、トポロジー３０５と類似の構成をもち、（第１のユーザのミドルボックスＤと同じタイプの）ミドルボックスＨを含む論理ネットワークトポロジー５０５を入力する。加えて、第２のユーザは、第１のユーザと同じミドルボックス構成設定インタフェースを介して、ミドルボックス規則５１０を入力する。第２のユーザは第１のユーザとは異なる物理マシンでデータを入力するが、ミドルボックスタイプは同じである（同じ実装形を用いる）ので、これらのユーザはデータを同じインタフェースの異なるコピーを介して入力する。

【００７４】

図示のように、第２の論理コントローラ５１５が規則５２０を（例えば、データベーステーブルレコードとして）受信する。論理コントローラ５１５は以前の例のように、規則５２０を受信すべき特定のノードを識別する。この場合、ミドルボックス４２５とＭＳＥ３５０を備えるノードは、第１のネットワーク３０５と第２のネットワーク５０５の両方に対する仮想マシンに対するホストとなるので、そのノードを管理する同じ物理コントローラ３４５が規則４１５を受信する。

【００７５】

ミドルボックスＨはミドルボックスアプリケーション４２５の新しい仮想ミドルボックスの作成を必要とするので、物理コントローラ３４５はそのミドルボックスのインスタンスに対する新しい異なるスライシング識別子５２５を割当てる。それから、物理コントローラ３４５は規則５２５とスライシング識別子５２５をホストマシンのミドルボックス４２５に分配する。ミドルボックス４２５は新しいミドルボックスインスタンスを作成し、規則５２０を（物理コントローラ３４５から受信する他の構成設定規則とともに）新しく作成されたミドルボックスインスタンスに対する新しいセットのデータプレーン規則５３０に変換する。

【００７６】

加えて、ミドルボックス４２５は別のエントリをそのスライスバインディングテーブル４３５に付加する。上述のように、ミドルボックスは自分自身の内部識別子を作成し、物理コントローラ３４５からの割当てられたスライシングＩＤへのこの内部ＩＤのバインディングをスライスバインディングテーブル４３５に格納する。ＭＳＥ３５０からパケットを受信するときに、このようにして、ミドルボックス４２５はパケットを処理するための適切なセットのミドルボックス規則を用いるためにスライシング識別子を除去して、その識別子をその内部ＩＤと一致させることができる。加えて、ミドルボックスが（例えば、新しいＴＣＰ接続に対して）新しい状態を作成するとき、そのミドルボックスは内部識別子を用いてその状態を特定のインスタンスに関係づけする。

【００７７】

上述の例は分散型ミドルボックスに関するものである。集中型ミドルボックスに関して言えば、ユーザは同じ方法で（そのミドルボックスに対して設計されたインタフェースを介して）構成設定を入力する。そのミドルボックスを管理する単一の物理コントローラ（或いは、単一の仮想マシンとして実装されているなら、ミドルボックスが配置されるホストマシン）のみを識別し、この物理コントローラに構成設定規則をエクスポートする。集中型ミドルボックスもまた複数のネットワーク間で仮想化されるので、物理コントローラは、その構成設定に対するスライス識別子を割当てる。

【００７８】

分散型のケースでは、ただ１つの管理されるスイッチング要素はパケットを特有の分散ミドルボックス要素を送信する一方、集中型のケースでは、多数の異なるＭＳＥはスライシング識別子を受信する必要がある。そのようなものとして、幾つかの実施例では、集中型ミドルボックスを管理する物理コントローラは特定のネットワークを管理する論理コントローラにスライシング識別子を送信する。その特定のネットワークは、このスライシング識別子を（例えば、ミドルボックスに宛てられたパケットにその識別子を付加するフローエントリの形式で）、ミドルボックスに（その管理物理コントローラを介して）パケットを送信する複数のノード全てに配信する。他の実施例では、論理コントローラはミドルボックスに対するスライシング識別子を割当て、この情報を管理物理コントローラを介して複数のノード全てに配信する。

【００７９】

加えて、ネットワーク制御システムは、集中型ミドルボックスとそのミドルボックスにパケットを送信する管理される種々のスイッチング要素との間のトンネルをセットアップする。トンネル情報は、入力変換コントローラインタフェースにより入力され、ミドルボックスによりサポートされる異なるトンネリングプロトコルの知識を備えた論理コントローラにより自動的に生成される。物理制御プレーンへの変換において、論理コントローラは、パケットからのトンネルによるカプセル化を付加したり除去したりするトンネルによるカプセル化のフローエントリを付加する。それから、これらのエントリは物理コントローラレベルでカスタマイズされ、異なる管理されるスイッチング要素各々で用いられる特定のポートとカプセル化とを考慮する。

【００８０】

ＩＩ．ネットワークコントローラアーキテクチュア

上述のセクションでは幾つかの異なるタイプのネットワークコントローラを含むネットワーク制御システムを説明した。図６は（例えば、論理コントローラや物理コントローラのような）ネットワークコントローラ６００のアーキテクチュアの例を図示している。幾つかの実施例のネットワークコントローラはテーブルマッピングエンジンを用いて複数のテーブルからなる入力の組からのデータを複数のテーブルからなる出力の組のデータにマッピングする。コントローラの複数のテーブルからなる入力の組は、論理転送プレーン（ＬＦＰ）データにマッピングされる論理制御プレーン（ＬＣＰ）データと、ユニバーサル物理制御プレーン（ＵＰＣＰ）データにマッピングされるＬＦＰデータと、カスタマイズされた物理制御プレーン（ＣＰＣＰ）データにマッピングされるＵＰＣＰデータとの内の少なくともいずれかを含む。また、複数のテーブルからなる入力の組は、別のコントローラに送信されるミドルボックス構成設定データと分散ミドルボックスインスタンスとの内の少なくともいずれかを含む。図示されるようにネットワークコントローラ６００は、入力テーブル６１５、規則エンジン６１０、出力テーブル６２０、インポータ６３０、エクスポータ６３５、変換器６３５、データ永久記憶装置（ＰＴＤ）６４０を含む。

【００８１】

幾つかの実施例では、入力テーブル６１５はネットワーク制御システムにおけるコントローラ６００の役割に依存して異なるタイプの出力をもつ複数のテーブルを含む。例えば、コントローラ６００がユーザの論理転送要素のために論理コントローラとして機能するとき、入力テーブル６１５はその論理転送要素のためのＬＣＰデータとＬＦＰデータとを含む。コントローラ６００が物理コントローラとして機能するとき、入力テーブル６１５はＬＦＰデータを含む。入力データ６１５はまた、ユーザ又は別のコントローラから受信するミドルボックス構成設定データを含む。そのミドルボックス構成設定データはミドルボックスが組み込まれる論理スイッチング要素を識別する論理データパスセットパラメータに関係している。

【００８２】

入力テーブル６１５に加えて、制御アプリケーション６００は、そのテーブルマッピング操作のための入力を収集するために規則エンジン６１０が用いるその他もろもろのテーブル（不図示）を含む。これらその他のテーブルには規則エンジン６１０がそのテーブル

マッピング操作を実行するのに必要とする定数のために規定された値（例えば、値０、再実行のためのディスパッチポート番号など）を格納する定数テーブルを含む。その他のテーブルはさらに、規則エンジン６１０が値を計算して出力テーブル６２５に投入するために用いる関数を格納する関数テーブルを含む。

【００８３】

規則エンジン６１０は入力データを出力データに変換するための１つの方法を特定するテーブルマッピング操作を実行する。複数の入力テーブルの１つが変形される（入力テーブルイベントという）ときにはいつでも、規則エンジンは１つ以上の出力テーブルにおける１つ以上のデータの組の変型をもたらすことになるかもしれない複数のテーブルマッピング操作の組を実行する。

【００８４】

幾つかの実施例では、規則エンジン６１０はイベントプロセッサ（不図示）、複数の問い合わせ方法（不図示）、テーブルプロセッサ（不図示）を含む。各問い合わせ方法は、入力テーブルイベント発生時に実行されることになる複数の統合操作からなる組を指定する複数の規則からなる組である。規則エンジン６１０のイベントプロセッサは、そのようなイベントの発生を検出する。いくつの実施例では、イベントプロセッサは入力テーブル６１５におけるレコードに変更通知用の入力テーブルを備えたコールバックの登録を行い、そのレコードの１つが変更されたとき、入力テーブルからの通知を受信することにより入力テーブルイベントを検出する。

【００８５】

検出された入力テーブルイベントに応じて、イベントプロセッサは、（１）検出されたテーブルイベントについての適切な問い合わせ方法を選択し、（２）その問い合わせ方法を実行するためにテーブルプロセッサに指示を与える。その問い合わせ方法を実行するために、幾つかの実施例では、テーブルプロセッサは、１つ以上の入力テーブルなどからの複数のデータ値からなる１つ以上の組を表現する１つ以上のレコードを生成するために、その問い合わせ方法により特定される統合操作を実行する。それから、幾つかの実施例のテーブルプロセッサでは、（１）選択動作を実行して統合操作により生成されるレコードからの複数のデータ値からなるサブセットを選択し、（２）１つ以上の出力テーブル６２０に複数の値からなる選択されたサブセットを書込む。

【００８６】

幾つかの実施例では、データログデータベース言語の変化を利用して、アプリケーション開発者がコントローラ用の規則エンジンを作成し、これにより、コントローラが論理データパスセットを制御される物理スイッチングインフラストラクチュアにマッピングする方法を特定できるようにする。データログデータベース言語の変化はここではｎＬｏｇとして言及される。データログのように、ｎＬｏｇは、異なるイベントの発生時に実行されることになる異なる動作を開発者が特定できるようにする幾つかの宣言的な規則と演算子とを提供する。幾つかの実施例では、ｎＬｏｇは、ｎＬｏｇの動作速度を向上させるためにデータログにより備えられる複数の演算子からなる限定的なサブセットを提供する。例えば、幾つかの実施例では、ｎＬｏｇは複数の宣言的な規則のいずれかでＡＮＤ演算子が用いられるのを許可するだけである。

【００８７】

ｎＬｏｇにより特定される宣言的な規則と演算子とは、それから、ｎＬｏｇコンパイラにより、より大きな規則の組へとコンパイルされる。幾つかの実施例では、このコンパイラはイベントをアドレスすることを意味する各規則をデータベースの統合操作のいくつかの組へと変換する。集合的には、より大きな組の規則は、ｎＬｏｇエンジンとして言及されるテーブルマッピング規則エンジンを形成する。

【００８８】

幾つかの実施例では、入力イベントが論理データパスセットパラメータに基づくものであるように規則エンジンにより実行される第１の統合操作を指定する。この指定は、その規則エンジンがコントローラにより管理されない論理データパスセット（即ち、論理ネッ

トワーク）に関係する統合操作の組を開始したときに、規則エンジンの統合操作が行えず、すぐさま終了することを保証する。

【００８９】

入力テーブル６１５のように、入力テーブル６２０は、コントローラ６００の役割に依存して異なるタイプのデータを備える複数のテーブルを含む。コントローラ６００が論理コントローラとして機能するとき、出力テーブル６１５は論理スイッチング要素に関してＬＦＰデータとＵＰＣＰデータとを含む。コントローラ６００が物理コントローラとして機能するとき、出力テーブル６２０はＣＰＣＰデータを含む。入力テーブルのように、出力テーブル６１５は、コントローラ６００が物理コントローラとして機能するとき、スライス識別子を含むことができる。

【００９０】

幾つかの実施例では、複数の出力テーブル６２０は幾つかのカテゴリへとグループ化される。例えば、幾つかの実施例では、複数の出力テーブル６２０は、規則エンジン（ＲＥ）入力テーブルとＲＥ出力テーブルとの内の少なくともいずれかであり得る。出力テーブルは、出力テーブルの変更により規則エンジンが問い合わせ方法の実行を要求する入力イベントを検出するようになるとき、ＲＥ入力テーブルである。出力テーブルは、規則エンジンが別の問い合わせ方法を実行するようにさせるイベントを生成するＲＥ入力テーブルとになり得る。出力テーブルは、出力テーブルの変更によりエクスポータ６２５がその変更を他のコントローラ又はＭＳＥにエクスポートさせるようにするとき、ＲＥ出力テーブルである。出力テーブルは、ＲＥ入力テーブル、ＲＥ出力テーブル、或いは、ＲＥ入力テーブルとＲＥ出力テーブルの両方にもなり得る。

【００９１】

エクスポータ６２５は、出力テーブル６２０のＲＥ出力テーブルの変更を検出する。幾つかの実施例では、エクスポータはＲＥ出力テーブルのレコードに変更通知用のＲＥ出力テーブルを備えたコールバックの登録を行う。幾つかの実施例では、エクスポータ６２５は、そのレコードの１つが変更されたとき、ＲＥ出力テーブルからの通知を受信するときに出力テーブルイベントを検出する。

【００９２】

検出された出力テーブルイベントに応じて、エクスポータ６２５は変形されたＲＥデータベースにおける各変形されたデータの組をとり、この変形されたデータの組を１つ以上の他のコントローラ又は１つ以上のＭＳＥに伝搬する。出力テーブルのレコードを別のコントローラに送信するとき、エクスポータは幾つかの実施例では単一の通信チャネル（例えば、ＲＰＣチャネル）を用いてそのレコードに含まれるデータを送信する。ＲＥ出力テーブルのレコードをＭＳＥに送信するとき、エクスポータは幾つかの実施例では２つのチャネルを用いる。１つのチャネルはＭＳＥの制御プレーンへフローエントリを書込むためにスイッチ制御プロトコル（例えば、0penFlow）を用いて確立される。もう１つのチャネルはデータベース通信チャネルプロトコル（例えば、ＪＳＯＮ）を用いて確立され構成設定データ（例えば、ポート構成設定、トンネル情報）を送信する。

【００９３】

幾つかの実施例では、コントローラ６００は出力テーブル６２０に、コントローラが管理を担当しない論理データパスセット（即ち、他の論理コントローラにより管理される論理ネットワーク）についてのデータを保持する。しかしながら、そのようなデータは変換器６３５によりＰＴＤ６４０に格納されるフォーマットへと変換され、それからＰＴＤに格納される。ＰＴＤ６４０はこのデータを１つ以上の他のコントローラのＰＴＤへと伝搬し、論理データパスセットの管理を担当する他のコントローラがデータを処理できるようにする。

【００９４】

幾つかの実施例では、そのコントローラはまた、出力テーブル６２０に格納されたデータをデータ復元のためにＰＴＤへと持ち込んでくる。それ故に、幾つかの実施例では、コントローラのＰＴＤはネットワーク制御システムにより管理される全ての論理データパス

セットについての全ての構成設定データをもつ。即ち、各ＰＴＤは全てのユーザの論理ネットワークの構成設定の全体概要を含む。

【００９５】

インポータ６３０は数多くの異なる発信元からの入力データとのインタフェースとなり、その入力データを用いて入力テーブル６１０を変形したり作成したりする。幾つかの実施例のインポータ６２０では、別のコントローラからの入力データを受信する。また、インポータ６２０はＰＴＤ６４０とのインタフェースとなり、他のコントローラインスタンスからＰＴＤを介して受信したデータが変換され、入力データとして使用され入力テーブル６１０を変形したり、作成したりできるようにする。さらにその上、インポータ６２０はまた、出力テーブル６３０におけるＲＥ入力テーブルでの変更を検出する。

【００９６】

ＩＩＩ．パケット処理

上記セクションではネットワーク制御システムを用いて論理ネットワークについてのフローエントリとミドルボックス構成設定の作成について詳細に説明した。それから、このデータはネットワークの物理的な実施形の中で（例えば、管理されるスイッチング要素におけるフローエントリに対するパケットをマッチングさせることにより）トラフィックを処理し転送するために用いられる。

【００９７】

図７は幾つかの実施例における第１の仮想マシンから第２の仮想マシンへパケットを送信するために実行する動作を概念的に図示している。図示のように、ＶＭ７０５は第１のホストマシン７１０に常駐し、また、第１のホストマシン７１０は管理されるスイッチング要素７１５とミドルボックス要素７２０とを含む。ＶＭ７２５は、第２のホストマシン７３０にＭＳＥ７３５（と不図示ではあるがミドルボックス要素）とともに常駐する。この例では、２つの仮想マシンが論理Ｌ３ルータにより接続される異なる論理Ｌ２ドメインに配置される。

【００９８】

ＶＭ７０５はパケットをＭＳＥ７１５に（例えば、ホストマシン７１０内で仮想インタフェースを介して）送信する。パケットはＶＭ７０５に対応する発信元ＭＡＣとＩＰアドレスと、ＶＭ７２５に対応する宛先ＩＰアドレス（と、利用可能なら宛先ＭＡＣアドレス）とをもつ。ＭＳＥ７１５は論理スイッチＡ（ＶＭ７２５がアタッチする論理スイッチ）に対するＬ２フローを実行することにより開始する。論理スイッチＡは、物理発信元ポート（仮想インタフェース）を論理発信元ポートにマッピングし、それから論理Ｌ２転送の決定を行ってパケットを論理ルータに送信することを含む。この論理ルータはまた、ＭＳＥ７１５により実現されるので、パケットの送信にはそのパケットの再実行が関与するだけである。

【００９９】

ＭＳＥ７１５はそれから論理ルータに対するＬ３フローを実行する。いくつかの実施例では、転送テーブルはパケットを、宛先ＩＰアドレスに基づいてＶＭ７２５がアタッチする論理スイッチＢに転送するためのフローエントリを含む。しかしながら、Ｌ３転送テーブルはまた、ユーザが入力したルーティングポリシーに基づいて（例えば、発信元ＩＰアドレスと論理発信元ポートとの内のいずれか、或いは、他のデータに基づいて）パケットをミドルボックスに転送するためのより高い優先度のエントリを含む。従って、Ｌ３転送の決定は、パケットをミドルボックスに送信することである。２つのソフトウェア要素の間でネゴシエートされたソフトウェアポートでパケットを送信する前に、ＭＳＥ７１５はスライスタグをパケットに付加して正しいミドルボックスインスタンスを識別する。幾つかの実施例では、ＭＳＥはこのタグを、パケットヘッダの特定のフィールドの先頭に付加する。

【０１００】

ミドルボックス７２０はパケットを受信して、スライスタグを除去してそのパケットを処理するために正しいミドルボックスインスタンスを識別する。（スライスバインディン

グテーブルを介して）識別された正しいインスタンスを用いて、ミドルボックスはその処理を実行する。このことには、（Ｓ－ＮＡＴに関する）発信元ＩＰを変形すること、（負荷バランサに関する）宛先ＩＰを変形すること、（ファイヤウォールに関して）パケットを削除するか許可するかを決定すること、或いは、他の処理などが関与する。パケットにおけるその処理を実行後、ミドルボックスはパケットを管理されるスイッチング要素へと戻す。

【０１０１】

幾つかの実施例では、ミドルボックスは新しいパケットをＭＳＥへと戻す。ＭＳＥは複数のミドルボックスのインスタンスから同じポートによりパケットを受信することができるので、幾つかの実施例では、ミドルボックスは、ソフトウェアポートによりパケットを送信する前に、これにスライスタグを付加する。ＭＳＥ７１５では、フローエントリはパケットを論理Ｌ３ルータ（即ち、ミドルボックスに接続されたＬ３ルータの発信元ポート）にマッピングするように戻し、それからＬ３転送を実行する。論理発信元ポートは今やミドルボックスに接続しているポートなので、ミドルボックスに送信するためのルーティングポリシーは実行されず、宛先ＩＰアドレスが論理スイッチＢに対する転送の決定となるという結果になる。このスイッチにおける論理転送の決定は、宛先ＭＡＣアドレスに基づくものであり、そのアドレスがＶＭ７２５への転送決定の結果となる。転送決定はパケットをカプセル化するために用いられ、また、（トンネルカプセル化を付加後）パケットが送信されるホストマシンの特定の物理ポートへとマッピングを行う。

【０１０２】

パケットはトンネルを介してネットワークを縦断し第２のホスト７３０のＭＳＥ７３５に到達する。トンネルを除去した後、ＭＳＥ７３５は宛先のコンテキストを読み出し、これを除去し、パケットをＶＭ７２５へと配信する。これはホストマシン７３０内で、パケットがＶＭ７２５に到達するように仮想インタフェースにマッピングを行う。

【０１０３】

当業者であれば、この例は分散ミドルボックスを含む数多くのあり得るパケット処理の例の１つに過ぎないことを理解するであろう。宛先ＶＭ７２５が発信元ＶＭ７０５とともにホスト７１０に位置していたなら、そのパケットはＭＳＥ７１５へ、ミドルボックス７２５へと進み、ＭＳＥ７１５に戻り、それから決して物理マシンを離れることなく宛先ＶＭ７１５へと達する。別の例として、もしミドルボックスが複製パケットを受信するだけであるなら、ＭＳＥ７１５はその受信パケットを第２のホスト７３０へ、複製パケットをミドルボックス７２０へと送信する。そのミドルボックスはその処理が終了後にはパケットを送信することはない。

【０１０４】

図７の例ではパケットを分散ミドルボックスに送信することに関与する処理を図示する一方、図８は複数のホストマシンのいずれかの外側に位置する集中型ミドルボックスを介したパケット送信を概念的に図示している。図示のように、この図ではホスト７１０のＶＭ７０５が再びホスト７３０のＶＭ７２５にパケットを送信する。発信元ＶＭ７０５からパケットを受信する際に、ＭＳＥ７１５はＬ２フロー（ＶＭ７０５に接続された論理発信元ポートへの発信元でのマッピングを行い、その後、論理Ｌ３ルータへの論理転送）を実行する。Ｌ３ルータでは、ルーティングポリシーは再びパケットを、この場合にはホスト７１０内部に配置されていないミドルボックスに送信する。そのようなものとして、ＭＳＥはスライスタグをパケットに（ミドルボックスが分散されているかのように同じ方法で）付加するが、物理ネットワークによりパケットを集中型ミドルボックス８０５に送信するためにトンネルカプセル化をパケットに加える。

【０１０５】

集中型ミドルボックス８０５はスライス識別子をその仮想ミドルボックスの１つに（即ち、そのスライスバインディングテーブルを用いて）マッピングし、それからミドルボックス処理を実行する。集中型ミドルボックスの例は、いくつかの実施例では、（分散化もあり得る）複数のファイヤウォール、（複製パケットを受信する受動型ミドルボックスで

ある）侵入検出システム、（ＷＬＡＮによりパケットを送信する前に種々のデータ圧縮技術を実行する）ＷＬＡＮオプティマイザーとともに、他の複数のミドルボックスを含む。そのミドルボックス処理の実行後、ミドルボックス８０５は新しいパケットをプールノードに送信する。

【０１０６】

いくつかの実施例では、集中型ミドルボックス各々はそのパケット全てを特定のプールノード（或いは、異なるミドルボックスインスタンスに対する異なるプールノード）に送信する。なぜなら、ミドルボックスはそのパケットを宛先に送信するのに必要な論理転送を実行する能力をもっていないことがあるかもしれないからである。従って、ミドルボックス８０５はトンネルのためにパケットをカプセル化し、（そして、いくらかの実施例ではそのパケットにスライスタグを付加し）、それから、これをプールノード８１０に送信する。プールノード８１０はパケットを正しい論理ルータにマッピングし、そのパケットの宛先ＩＰアドレスを用いて、宛先ＶＭ７２５が接続する論理スイッチに対する転送決定を行う。プールノードでの論理Ｌ２フローはパケットを宛先マシン７２５に転送し、この宛先コンテキストがパケットをカプセル化するのに用いられる。それから、プールノード８１０は物理ネットワークによるトランスポートのためにトンネルカプセル化を付加し、パケットをホスト７３０に送信する。ＭＳＥ７３５はこの宛先コンテキストを読み出し、これを除去して、（図７に示すのと同じ方法で）パケットをＶＭ７２５に配信する。

【０１０７】

図９は、管理されるスイッチング要素９１０を有する集中型ミドルボックス９０５の第２の例である。この場合、パケットは図８に示す前の例と同じ方法で集中型ミドルボックス９０５に到達し、ミドルボックス９０５は同じ処理を実行する。しかしながら、トンネルによる送信のために新しいパケットをカプセル化するというよりはむしろ、そのミドルボックスの処理が、内蔵したＭＳＥ９１０に（例えば、ホストマシン内で機能するものに類似のソフトウェアポートを介して）パケットを送信する。ＭＳＥ９１０は前の例のプールノードと同じ処理を実行し、実際にはパケットをＶＭ７２５に送信するための論理Ｌ２転送決定を行って、この宛先コンテキスト（とトンネルカプセル化）とともにパケットをカプセル化し、それから、そのパケットをネットワークによりホスト７３０に送信する。この点で、この処理は図８に示したものと同じになる。

【０１０８】

ＩＶ．電子システム

上述の特徴やアプリケーションの多くは、（コンピュータ可読媒体としても言及される）コンピュータ可読記憶媒体に記録された命令セットとして規定されるソフトウェア処理として実施される。これらの命令が１つ以上のコンピュータ又は処理ユニット（例えば、１つ以上のプロセッサ、複数のプロセッサのコア、或いは別の処理ユニットなど）により実行されるとき、それらの命令により処理ユニットがその命令で指示される動作を実行するようになる。コンピュータ可読媒体の例は、以下のものに限定されるものではないが、ＣＤ－ＲＯＭ、フラッシュドライブ、ランダムアクセスメモリ（ＲＡＭ）チップ、ハードドライブ、消去可能プログラム可能な読出専用メモリ（ＥＰＲＯＭ）、電気的消去可能プログラム可能な読出専用メモリ（ＥＥＰＲＯＭ）を含む。そのコンピュータ可読媒体は、搬送波、無線的に或いは有線接続で伝搬する電気信号を含むものではない。

【０１０９】

この明細書では、“ソフトウェア”という用語は、読出専用メモリに常駐するファームウェア、又は、プロセッサによる処理のためにメモリへと読み出される磁気記憶装置に格納されたアプリケーションを含むことを意味する。また、幾つかの実施例では、複数のソフトウェア発明が大きなプログラムの一部として、一方、残りは明白なソフトウェア発明として実施される。幾つかの実施例では、複数のソフトウェア発明が個々のプログラムとして実施される。最後に、ここで説明するソフトウェア発明を共に実施する個別のプログラムの何らかの組み合わせも本発明の範囲の中にある。幾つかの実施例では、１つ以上の電子システムで動作するためにインストールされるとき、そのソフトウェアプログラムは

そのソフトウェアプログラムを実行し、その動作を実行する１つ以上の具体的なマシンでの実施を規定する。

【０１１０】

図１０は本発明の幾つかの実施例が実装される電子システム１０００を概念的に図示している。電子システム１０００は、コンピュータ（例えば、デスクトップコンピュータ、パーソナルコンピュータ、タブレットコンピュータなど）、サーバ、専用スイッチ、ＰＤＡ、他の何らかの電子的又はコンピュータデバイスなどである。そのような電子システムは種々のタイプのコンピュータ可読媒体と、別のタイプの種々のコンピュータ可読媒体に対するインタフェースとを含む。電子システム１０００は、バス１００５、処理ユニット１０１０、システムメモリ１０２５、読出専用メモリ１０３０、永久記憶デバイス１０３５、入力デバイス１０４０、及び出力デバイス１０４５を含む。

【０１１１】

バス１００５は包括的に全てのシステム、周辺機器、電子システム１０００の数多くの内部デバイスを通信可能に接続するチップセットバスを表現している。例えば、バス１００５は通信可能に読出専用メモリ１０３０、システムメモリ１０２５、永久記憶デバイス１０３５に処理ユニット１０１０を接続している。

【０１１２】

これらの種々のメモリユニットから、処理ユニット１０１０は本発明の処理を実行するために、実行する命令と処理するデータを取り出す。その処理ユニットは、単一のプロセッサでも良いし、或いは、異なる実施例ではマルチコアのプロセッサでも良い。

【０１１３】

読出専用メモリ（ＲＯＭ）１０３０は、処理ユニット１０１０により必要とされる静的なデータと命令と、その電子システムの他のモジュールとを格納する。これに対して、永久記憶デバイス１０３５は、読出し書込みメモリである。このデバイスは、電子システム１０００がオフであるときでさえ、命令とデータを格納する不揮発性メモリユニットである。本発明の幾つかの実施例では、（磁気ディスク又は光ディスクとそれに対応するディスクドライブのような）大容量記憶装置を永久記憶デバイス１０３５として用いる。

【０１１４】

他の実施例では（フロッピィディスク（登録商標）、フラッシュメモリデバイスなど、そして、それに対応するディスクドライブのような）着脱可能な記憶デバイスを永久記憶デバイスとして用いる。永久記憶デバイス１０３５と同様に、システムメモリ１０２５は読出し書込みメモリである。しかしながら、記憶デバイス１０３５とは異なり、システムメモリ１０２５はランダムアクセスメモリのような、揮発性の読出し書込みメモリである。システムメモリ１０２５はプロセッサが実行時間に必要とする命令とデータとのいくらかを格納する。幾つかの実施例では、本発明の処理はシステムメモリ１０２５と永久記憶デバイス１０３５と読出専用メモリとの内の少なくともいずれかに格納される。これらの種々のメモリユニットから、処理ユニット１０１０は幾つかの実施例の処理を実行するために、実行する命令と処理するデータを取り出す。

【０１１５】

バス１００５はまた、入力デバイス１０４０と出力デバイス１０４５に接続する。入力デバイス１０４０によりユーザは情報とやり取りをして、電子システムに対する命令を選択することができる。入力デバイス１０４０は英数字キーボード、ポインティングデバイス（“カーソル制御デバイス”とも呼ばれる）、カメラ（例えば、ウェブカメラ）、マイクロフォン又は音声命令を受信する類似のデバイスなどを含む。出力デバイス１０４５は電子システム、さもなければ他の出力データにより生成される画像を表示する。出力デバイス１０４５はプリンタや、陰極線管（ＣＲＴ）や液晶ディスプレイ（ＬＣＤ）のような表示デバイスとともに、スピーカや類似のオーディオ出力デバイスを含む。他の実施例では、入力デバイスと出力デバイスの両方としての機能するタッチスクリーンのようなデバイスを含む。

【０１１６】

最後に、図１０に示すように、バス１０５はまた電子システム１０００をネットワークアダプタ（不図示）を介してネットワーク１０６５に結合する。このようにして、コンピュータは、（ローカルエリアネットワーク（“ＬＡＮ”）、広域エリアネットワーク（“ＷＬＡＮ”）、或いはイントラネット）のようなコンピュータのネットワークの一部、或いは、インターネットのような複数のネットワークの中のネットワークでも良い。電子システム１０００のいずれかの、又は、全ての構成要素が本発明と関連して用いられても良い。

【０１１７】

幾つかの実施例では、マイクロプロセッサ、記憶装置、機械読出可能なコンピュータ読出可能な媒体（或いは、コンピュータ可読記憶媒体、機械可読媒体、或いは、機械可読格納媒体として言及される）にコンピュータプログラム命令を格納するメモリのような電気的構成要素を含む。そのようなコンピュータ可読媒体のような例のいくつかは、ＲＡＭ、ＲＯＭ，読出専用コンパクトディスク（ＣＤ－ＲＯＭ）、書込み可能コンパクトディスク（ＣＤ－Ｒ）、再書込み可能なコンパクトディスク（ＣＤ－ＲＷ）、読出し専用多目的ディスク（例えば、ＤＶＤ－ＲＯＭ、デュアルレイヤＤＶＤ－ＲＯＭ）、種々の書込み可能な／再書込み可能なＤＶＤ（例えば、ＤＶＤ－ＲＡＭ、ＤＶＤ－ＲＷ、ＤＶＤ＋ＲＷ等）、フラッシュメモリ（例えば、ＳＤカード、ミニＳＤカード、マイクロＳＤカード等）、磁気及び／或いはソリッドステートハードドライブ、読出専用と書込み可能BluRay（登録商標）ディスク、超高密度光ディスク、別の何らかの光／磁気ディスク、フロッピィディスク（登録商標）を含む。コンピュータ可読媒体は、少なくとも１つの処理ユニットにより実行可能であり、種々の動作を実行する命令セットを含むコンピュータプログラムを格納することができる。コンピュータプログラム、或いは、コンピュータコードの例は、コンパイラにより生成されるようなマシンコードと、コンピュータ、電子的構成要素、インタプリタを用いるマイクロプロセッサにより実行される高レベルコードを含むファイルを含む。

【０１１８】

上記の検討は主としてソフトウェアを実行するマイクロプロセッサ或いはマルチコアプロセッサに言及しているが、幾つかの実施例は、アプリケーション専用集積回路（ＡＳＩＣ）或いはフィールドプログラマブルゲートアレイ（ＦＰＧＡ）のような１つ以上の集積回路により実行される。幾つかの実施例では、そのような集積回路はその回路自身に格納される命令を実行する。加えて、幾つかの実施例は、プログラム可能なロジックデバイス（ＰＬＤ）、ＲＯＭ、或いは、ＲＡＭデバイスに格納されるソフトウェアを実行する。

【０１１９】

この明細書とこの出願のいずれかの請求項で用いられるように、“コンピュータ”、“サーバ”、“プロセッサ”、及び、“メモリ”という用語の全ては電子機器、或いは、他の技術のデバイスに言及したものである。これらの用語は、人々、人々のグループを除外するものである。この明細書の目的のために、「表示」或いは「表示する」という用語は、電子機器上での表示を意味する。この明細書とこの出願のいずれかの請求項で用いられるように、“コンピュータ可読媒体”、“複数のコンピュータ可読媒体”、及び、“機械可読媒体”という用語は全く、コンピュータにより読出し可能な形式で情報を格納する有形の物理的な対象に限定されるものである。これらの用語は、無線信号、有線のダウンロード信号、及び、他の何らかの瞬間的な信号を排除するものである。

【０１２０】

本発明は数多くの具体的な詳細を参照して説明したが、当業者であれば本発明が本発明の精神を逸脱することなく他の具体的な形式で実施可能なものであることを認識するであろう。従って、当業者であれば、本発明が前述の例示的な詳細により限定されるものではなく、むしろ、添付の請求の範囲により規定されるべきものであることを理解するであろう。

【図１】

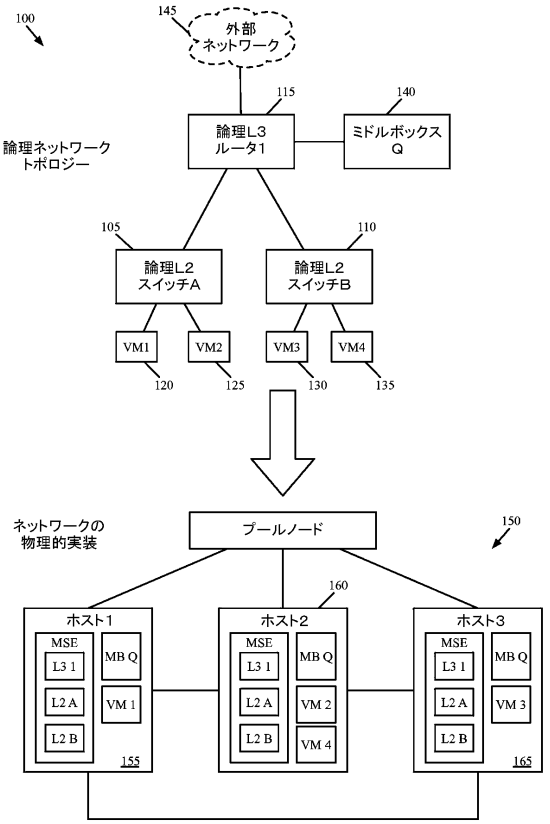

Figure 1

【図２】

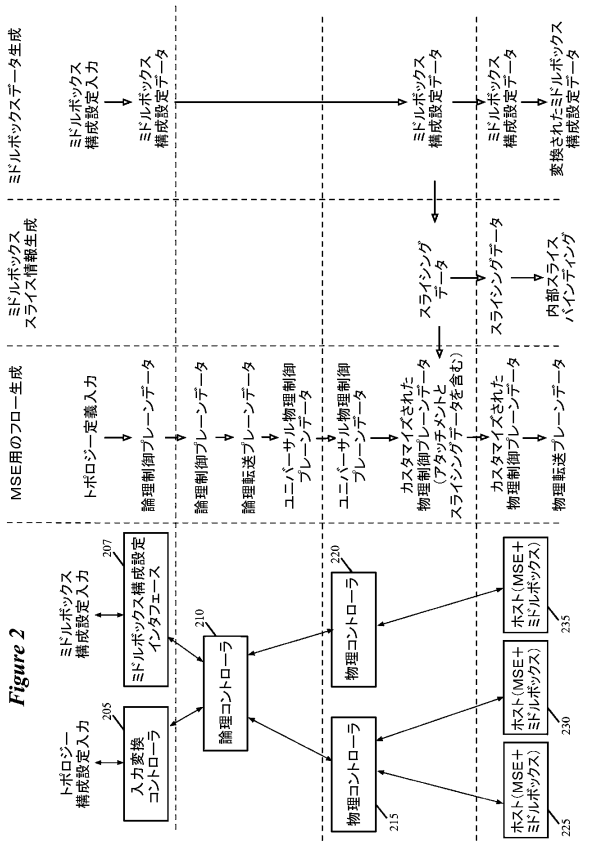

Figure 2

【図３】

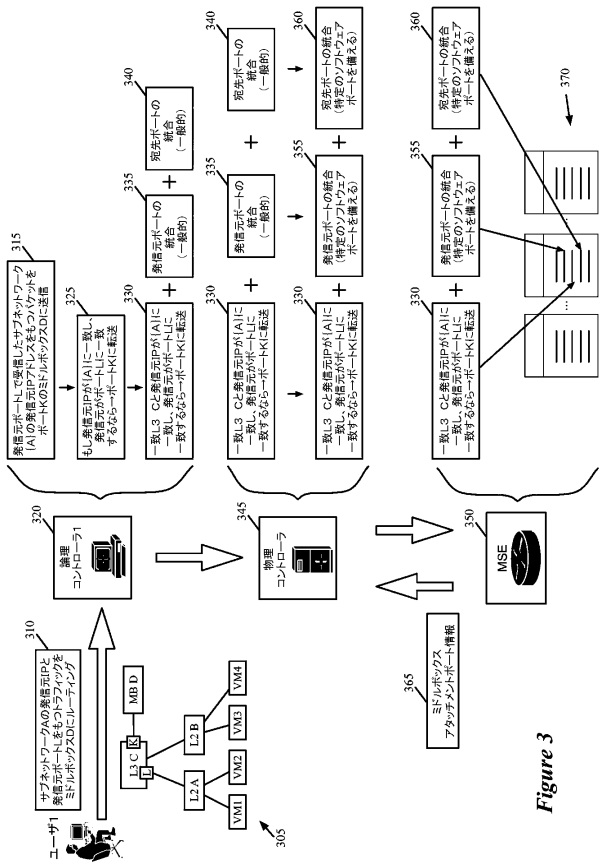

Figure 3

【図４】

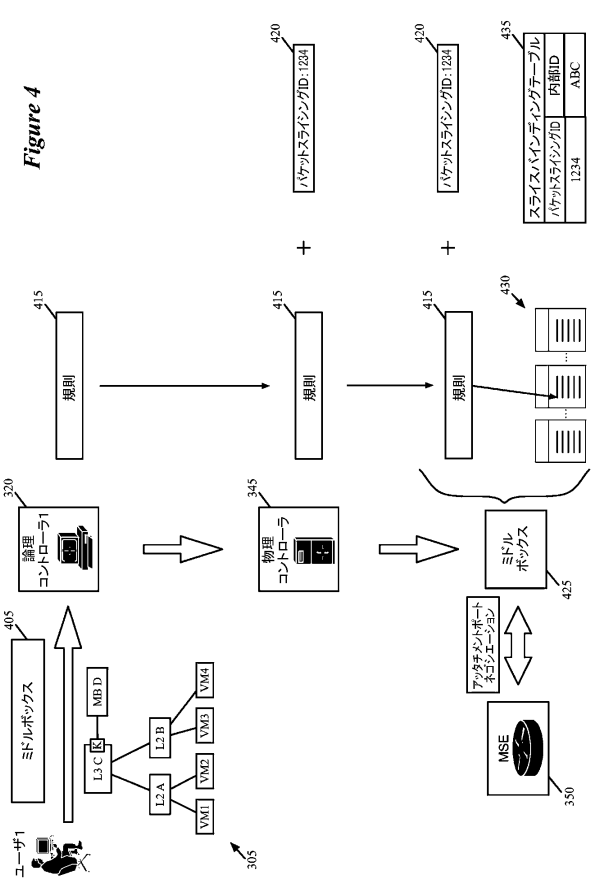

Figure 4

【図５】

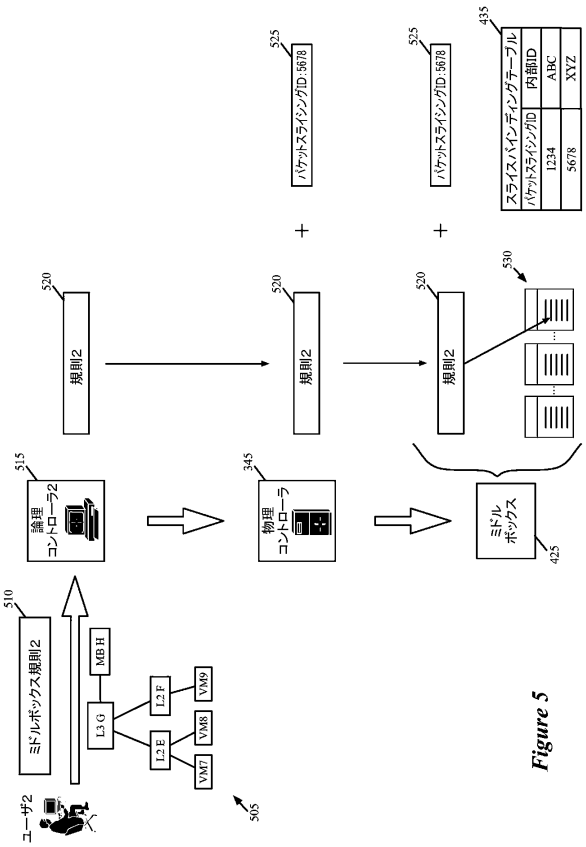

Figure 5

【図６】

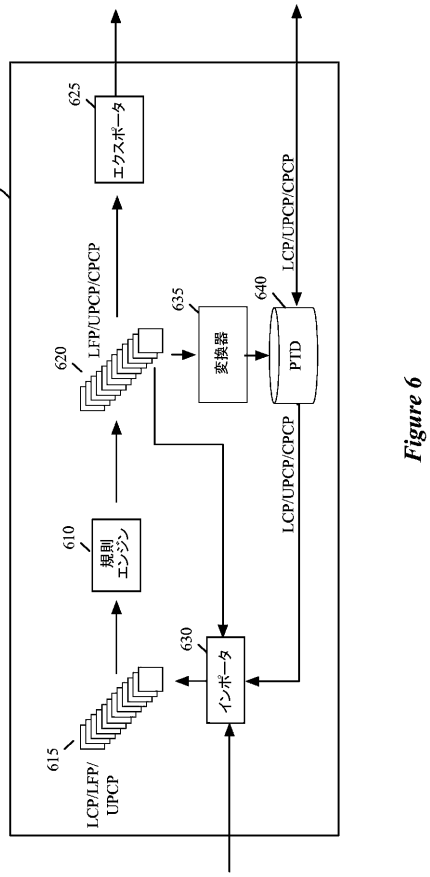

Figure 6

【図７】

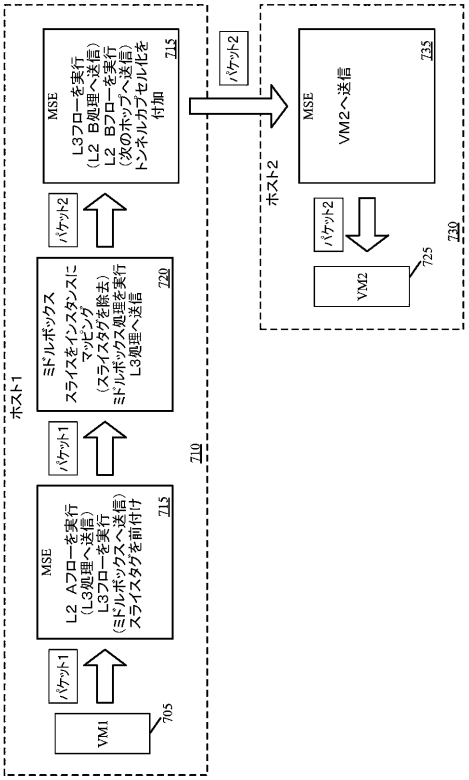

Figure 7

【図８】

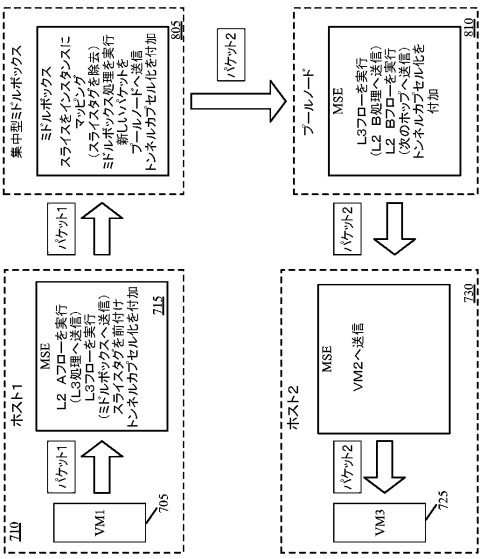

Figure 8

【図９】

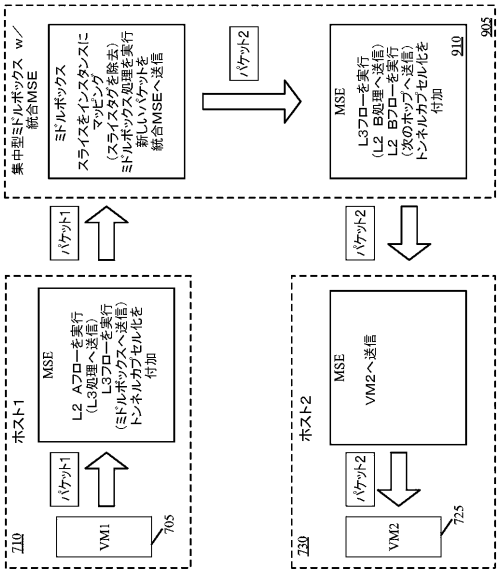

Figure 9

【図１０】

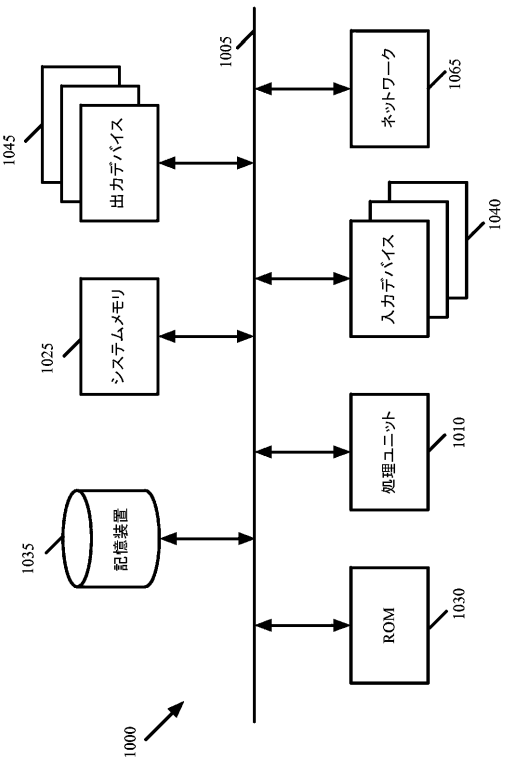

Figure 10

フロントページの続き

(72)発明者　チャン，　ロンフア
アメリカ合衆国　カリフォルニア州　９４３０４，　パロアルト，　ヒルビュー　アベニュー　３４０１
(72)発明者　コポネン，　テーム
アメリカ合衆国　カリフォルニア州　９４３０４，　パロアルト，　ヒルビュー　アベニュー　３４０１
(72)発明者　タッカー，　パンカジ
アメリカ合衆国　カリフォルニア州　９４３０４，　パロアルト，　ヒルビュー　アベニュー　３４０１
(72)発明者　パドマナバン，　アマー
アメリカ合衆国　カリフォルニア州　９４３０４，　パロアルト，　ヒルビュー　アベニュー　３４０１
(72)発明者　カサド，　マーティン
アメリカ合衆国　カリフォルニア州　９４３０４，　パロアルト，　ヒルビュー　アベニュー　３４０１

審査官　大石　博見

(56)参考文献　国際公開第２０１０／１１６６０６（WO，Ａ１）
特開２０１１－１８８４３３（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｈ０４Ｌ　１２／７０