

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第5849162号
(P5849162)

(45) 発行日 平成28年1月27日 (2016. 1. 27)

(24) 登録日 平成27年12月4日 (2015. 12. 4)

(51) Int. Cl. F I
H O 4 L 12/713 (2013.01) H O 4 L 12/713

請求項の数 15 (全 33 頁)

(21) 出願番号	特願2014-556838 (P2014-556838)	(73) 特許権者	511235548
(86) (22) 出願日	平成25年4月18日 (2013. 4. 18)		ニシラ, インコーポレイテッド
(65) 公表番号	特表2015-507448 (P2015-507448A)		アメリカ合衆国 カリフォルニア州 94
(43) 公表日	平成27年3月5日 (2015. 3. 5)		304, パロアルト, ヒルビュー ア
(86) 国際出願番号	PCT/US2013/037232		ベニユー 3401
(87) 国際公開番号	W02013/158918	(74) 代理人	100076428
(87) 国際公開日	平成25年10月24日 (2013. 10. 24)		弁理士 大塚 康德
審査請求日	平成26年8月12日 (2014. 8. 12)	(74) 代理人	100112508
(31) 優先権主張番号	61/635, 226		弁理士 高柳 司郎
(32) 優先日	平成24年4月18日 (2012. 4. 18)	(74) 代理人	100115071
(33) 優先権主張国	米国 (US)		弁理士 大塚 康弘
(31) 優先権主張番号	61/647, 516	(74) 代理人	100116894
(32) 優先日	平成24年5月16日 (2012. 5. 16)		弁理士 木村 秀二
(33) 優先権主張国	米国 (US)	(74) 代理人	100130409
			弁理士 下山 治

最終頁に続く

(54) 【発明の名称】 分散ネットワーク制御システムにおけるチャーン化を最小限にするためのトランザクションの使用

(57) 【特許請求の範囲】

【請求項 1】

ネットワークにおいてデータを転送する複数の被管理転送要素を含む前記ネットワークを管理する特定の制御器における方法であって、

第1の制御器から第1の入力セットを受信し、第2の制御器から第2の入力セットを受信する工程と、

前記第1の入力セットを使用して出力セットの計算を開始する一方で、前記第2の入力セットを格納する工程と、

前記第1の制御器の障害発生後、第3の入力セットの前記第2の制御器からの受信を開始する工程であって、前記第3の入力セットと、前記第1の入力セットまたは前記第2の入力セットとは、別の入力グループとは独立にまとめて処理されるべき特定の入力グループの一部である工程と、

前記第2の入力セット及び前記第3の入力セットを使用して、前記出力セットを計算する工程と、

前記特定の入力グループの前記第3の入力セットにおける最後の入力を前記特定の制御器が受信したことを示し、それにより前記特定の入力グループの全ての入力が前記特定の制御器に到達したことを示すインジケータを前記第2の制御器から受信する工程と、

前記インジケータを受信し前記出力セットを完全に計算した後に、第3の制御器に前記出力セットを送出する工程と

を備え、

10

20

前記第 3 の制御器は、その後に前記特定の制御器からの前記出力セットを処理し、前記処理された出力を被管理転送要素に送出することを特徴とする方法。

【請求項 2】

前記第 2 の入力セット内の少なくとも 1 つの入力は、前記第 1 の入力セット内の入力と重複し、

前記出力セットを計算する工程では、前記重複する入力が前記出力セットに影響を及ぼさないように前記出力セットを計算する工程を含むことを特徴とする請求項 1 記載の方法。

【請求項 3】

前記インジケータは、前記第 3 の入力セット内の入力の一部であることを特徴とする請求項 1 記載の方法。

【請求項 4】

前記第 1 の入力セット、前記第 2 の入力セット又は前記第 3 の入力セット内の入力はデータタプルであることを特徴とする請求項 1 記載の方法。

【請求項 5】

前記被管理転送要素に送出される前記処理された出力セットは、前記被管理転送要素の転送挙動を定義することを特徴とする請求項 1 記載の方法。

【請求項 6】

少なくとも 1 つの処理部によって実行される、ネットワークにおいてデータを転送する複数の被管理転送要素を含む前記ネットワークを管理する特定の制御器のためのプログラムを格納した非一時的機械読出可能媒体であって、

前記プログラムは、

第 1 の制御器から第 1 の入力セットを受信し、第 2 の制御器から第 2 の入力セットを受信する工程と、

前記第 1 の入力セットを使用して出力セットの計算を開始する一方で、前記第 2 の入力セットを格納する工程と、

前記第 1 の制御器の障害発生後、第 3 の入力セットの前記第 2 の制御器からの受信を開始する工程であって、前記第 3 の入力セットと、前記第 1 の入力セットまたは前記第 2 の入力セットとは、別の入力グループとは独立にまとめて処理されるべき特定の入力グループの一部である工程と、

前記第 2 の入力セット及び前記第 3 の入力セットを使用して、前記出力セットを計算する工程と、

前記特定の入力グループの前記第 3 の入力セットにおける最後の入力を前記特定の制御器が受信したことを示し、それにより前記特定の入力グループの全ての入力が前記特定の制御器に到達したことを示すインジケータを前記第 2 の制御器から受信する工程と、

前記インジケータを受信し前記出力セットを完全に計算した後に、第 3 の制御器に前記出力セットを送出する工程と

を実行するための命令の組を有し、

前記第 3 の制御器は、その後に前記特定の制御器からの前記出力セットを処理し、前記処理された出力を被管理転送要素に送出することを特徴とする非一時的機械読出可能媒体

【請求項 7】

前記第 2 の入力セット内の少なくとも 1 つの入力は、前記第 1 の入力セット内の入力と重複し、

前記出力セットを計算するための命令の組は、前記重複する入力が前記出力セットに影響を及ぼさないように前記出力セットを計算するための命令の組を含むことを特徴とする請求項 6 記載の非一時的機械読出可能媒体。

【請求項 8】

前記インジケータは、前記第 3 の入力セット内の入力の一部であることを特徴とする請求項 6 記載の非一時的機械読出可能媒体。

【請求項 9】

前記第 1 の入力セット、前記第 2 の入力セット又は前記第 3 の入力セット内の入力はデータタプルであることを特徴とする請求項 6 記載の非一時的機械読出可能媒体。

【請求項 10】

前記被管理転送要素に送出される前記処理された出力セットは、前記被管理転送要素の転送挙動を定義することを特徴とする請求項 6 記載の非一時的機械読出可能媒体。

【請求項 11】

ネットワークにおいてデータを転送する複数の被管理転送要素を含む前記ネットワークを管理する第 1 の制御器であって、

一組の処理部と、

前記一組の処理部による実行のためのプログラムを格納した非一時的機械読出可能媒体とを備え、

前記プログラムは、前記第 1 の制御器に、

前記第 1 の制御器から第 1 の入力セットを受信し、第 2 の制御器から第 2 の入力セットを受信する工程と、

前記第 1 の入力セットを使用して出力セットの計算を開始する一方で、前記第 2 の入力セットを格納する工程と、

前記第 1 の制御器の障害発生後、第 3 の入力セットの前記第 2 の制御器からの受信を開始する工程であって、前記第 3 の入力セットと、前記第 1 の入力セットまたは前記第 2 の入力セットとは、別の入力グループとは独立にまとめて処理されるべき特定の入力グループの一部である工程と、

前記第 2 の入力セット及び前記第 3 の入力セットを使用して、前記出力セットを計算する工程と、

前記特定の入力グループの前記第 3 の入力セットにおける最後の入力を前記特定の制御器が受信したことを示し、それにより前記特定の入力グループの全ての入力が前記特定の制御器に到達したことを示すインジケータを前記第 2 の制御器から受信する工程と、

前記インジケータを受信し前記出力セットを完全に計算した後に、第 3 の制御器に前記出力セットを送出する工程とを実行させるための命令の組を有し、

前記第 3 の制御器は、その後に前記特定の制御器からの前記出力セットを処理し、前記処理された出力を被管理転送要素に送出することを特徴とする第 1 の制御器。

【請求項 12】

前記第 2 の入力セット内の少なくとも 1 つの入力は、前記第 1 の入力セット内の入力と重複し、

前記出力セットを計算するための命令の組は、前記重複する入力が前記出力セットに影響を及ぼさないように前記出力セットを計算するための命令の組を含むことを特徴とする請求項 11 記載の第 1 の制御器。

【請求項 13】

前記インジケータは、前記第 3 の入力セット内の入力の一部であることを特徴とする請求項 11 記載の第 1 の制御器。

【請求項 14】

前記第 1 の入力セット、前記第 2 の入力セット又は前記第 3 の入力セット内の入力はデータタプルであることを特徴とする請求項 11 記載の第 1 の制御器。

【請求項 15】

前記被管理転送要素に送出される前記処理された出力セットは、前記被管理転送要素の転送挙動を定義することを特徴とする請求項 11 記載の第 1 の制御器。

【発明の詳細な説明】

【技術分野】

【0001】

10

20

30

40

50

本発明は、分散ネットワーク制御システムにおけるチャーン化を最小限にするためのトランザクションの使用に関する。

【背景技術】

【0002】

現在の企業の多くは、種々の接続、アプリケーション及びシステムをサポートするスイッチ、ハブ、ルータ、サーバ、ワークステーション及び他のネットワーク装置を含む大規模な高度ネットワークを有する。仮想マシンの移動、動的な作業負荷、マルチテナンシ、並びに顧客別のサービス品質及びセキュリティ構成を含むコンピュータのネットワーク化の更なる高度化により、それに対応できるネットワーク制御システムが必要とされる。これらの大規模な高度ネットワークを分散して処理するために、分散ネットワーク制御システムが提供されている。しかし、多くの場合、分散ネットワーク制御システムの1つの構成要素により行われたネットワーク状態の変更がシステムの残りの部分を通じて前後に波及することにより、分散ネットワーク制御システムにおいてチャーン化が生じる。

10

【発明の概要】

【0003】

本発明のいくつかの実施形態は、複数のネットワーク制御器により形成される階層の上位レイヤの第1の制御器及び第2の制御器から入力を受信する特定のネットワーク制御器を提供する。特定の制御器は、第1の制御器及び第2の制御器からの入力を処理し、第1の制御器からの入力のみを処理することにより生成された場合の出力と同一になるように出力を生成する。

20

【0004】

特に、いくつかの実施形態の特定の制御器は、第1の制御器から第1の入力セットを受信し、第2の制御器から第2の入力セットを受信する。その後、特定の制御器は、第1の入力セットを使用して出力セットの計算を開始する。第1の制御器の障害発生後、特定の制御器は、第2の制御器から第3の入力セットを受信する。第3の入力セットと第1の入力セット又は第2の入力セットとは、まとめて処理され且つ別の入力グループとは別に処理される入力グループを形成する。

【0005】

その後、特定の制御器は、入力グループの全ての入力が特定の制御器に到着したことを示すインジケータを第2の制御器から受信する。特定の制御器は、インジケータを受信し且つ出力セットを完全に計算した後、第4の制御器又は被管理転送要素に出力セットを送出する。その後、第4の制御器は、特定の制御器からの出力セットを処理し、処理された出力を被管理転送要素に送出する。

30

【0006】

本発明のいくつかの実施形態は、階層における上位レイヤの複数の異なる制御器の各々から入力を受信する階層の中間レイヤのネットワーク制御器を更に提供する。上位レイヤの制御器からの入力は、複数の異なるトランザクションとして供給される。いくつかの実施形態において、下位レイヤの制御器は、異なる制御器から受信した入力から出力を生成し、生成された出力を単一のトランザクションとして階層において更に下位のレイヤの制御器のセットに出力する。

40

【0007】

詳細には、中間レイヤのネットワーク制御器は、上位レイヤのネットワーク制御器のセットから複数の入力グループを受信する。各入力グループは、まとめて処理され且つ別の入力グループとは別に処理される。入力グループが特定の条件を満たす場合、中間レイヤのネットワーク制御器は、入力グループのうちの2つ以上をまとめて処理して出力セットを生成する。入力グループが特定の条件を満たさない場合、ネットワーク制御器は、1つの入力グループ毎にまとめて処理することにより入力グループを処理して出力セットを生成する。その後、ネットワーク制御器は、下位レイヤの制御器のセットに生成された出力セットを送出する。

【0008】

50

上記の発明の概要は、本発明のいくつかの実施形態の簡単な説明であることを意図する。これは、本明細書において開示される全ての発明の主題の説明又は概要であることを意図しない。以下の発明を実施するための形態及びそこで参照される図面により、発明の概要において説明した実施形態及び他の実施形態を更に説明する。従って、本明細書により説明する全ての実施形態を理解するためには、発明の概要、発明を実施するための形態及び図面を全て読む必要がある。更に、請求項は、発明の主題の主旨から逸脱せずに他の特定の形態で実現可能であるため、発明の概要、発明を実施するための形態及び図面における例示的な詳細により限定されず、添付の特許請求の範囲により定義される。

【図面の簡単な説明】

【0009】

10

本発明の新規の特徴を添付の特許請求の範囲に記載する。しかし、説明のために、本発明の複数の実施形態を以下の図面において記載する。

【図1】図1は、ネットワーク制御器の階層の例を示す図である。

【図2】図2は、いくつかの実施形態のネットワーク制御器のアーキテクチャを示す図である。

【図3】図3は、論理制御器から入力を受信する物理制御器を概念的に示す図である。

【図4】図4は、ネットワーク制御器の階層における上位レイヤに存在する送出元制御器のフェイルオーバを処理するためにいくつかの実施形態が実行する処理を概念的に示す図である。

【図5】図5は、論理制御器から入力を受信する物理制御器を概念的に示す図である。

20

【図6】図6は、複数の論理制御器から入力の変更を受信する物理制御器を概念的に示す図である。

【図7】図7は、複数のトランザクションを形成する入力の変更からトランザクションの出力の変更のセットを生成するためにいくつかの実施形態が実行する処理を概念的に示す図である。

【図8】図8は、ネットワーク制御器がユーザからの要求を被管理転送要素に分散し且つ要求に対する応答をユーザに返すネットワーク制御システムを示す図である。

【図9】図9は、物理制御器のセットから受信した汎用応答を集約するいくつかの実施形態の論理制御器を示す図である。

【図10】図10は、制御器の階層における下位レイヤの下位制御器からの応答のセットを集約して階層における上位レイヤの上位制御器に渡す単一の応答を生成するためにいくつかの実施形態が実行する処理を概念的に示す図である。

30

【図11】図11は、本発明のいくつかの実施形態が実現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】

【0010】

以下の本発明の詳細な説明において、本発明の多くの詳細、例及び実施形態を記載及び説明する。しかし、本発明は記載される実施形態に限定されず、説明する特定の詳細及び例のいくつかを用いずに実施されてもよいことが当業者には明らかになるだろう。

【0011】

40

いくつかの実施形態は、ネットワーク制御器が被管理転送要素のセットの転送挙動を定義するために被管理転送要素のセットにプッシュする転送状態情報を計算するネットワーク制御システムを提供する。いくつかの実施形態において、ネットワーク制御器は、複数の制御器レイヤを有する階層を形成する。論理制御器のセットは、階層の最上位レイヤに位置し、入力された論理制御プレーンデータから汎用物理制御プレーンデータを生成する。論理制御器のセットの下位レイヤは物理制御器のセットであり、いくつかの実施形態において、これは汎用物理制御プレーンデータを被管理転送要素に特有の物理制御プレーンデータにカスタマイズする。

【0012】

いくつかの実施形態において、物理制御器は、被管理転送要素に対するカスタマイズを

50

実際に実行するシャーシ制御器のセットに汎用物理制御プレーンデータを中継する。これらの実施形態において、シャーシ制御器は、制御器により形成される階層の最下位レイヤに存在する。物理制御器又はシャーシ制御器は被管理転送要素とインタフェースし、被管理転送要素にカスタマイズ済物理制御プレーンデータを供給する。被管理転送要素は、制御器から受信したデータを使用して、ネットワークにおいてデータを転送する。

【0013】

階層の上位レイヤの特定の制御器は、階層における下位レイヤの別の制御器に自身の出力データを供給する。いくつかの実施形態において、特定の制御器は、特定の制御器に対するホットスタンバイ又は冗長制御器として動作する（例えば、階層の下位レイヤの制御器に同一の出力データを供給する）バックアップ制御器を同一レイヤに有する。いくつかの実施形態において、下位レイヤの制御器は、特定の制御器から受信した出力データから自身の出力を生成する。

10

【0014】

特定の制御器に障害が発生した場合、下位レイヤの制御器は、バックアップ制御器からの同一の出力データを処理することで出力データが変化することのないように、（１）特定の制御器からそれまでに受信した出力データ及び（２）バックアップ制御器からの出力データから自身の出力データを生成する。すなわち、特定の制御器の障害発生後、下位レイヤの制御器は、障害発生前に特定の制御器から受信したデータと同一のデータを含む出力データをバックアップ制御器から受信し且つ処理する。しかし、下位レイヤの制御器は、下位レイヤの制御器の出力データが特定の制御器からの出力データのみを処理することにより生成された場合の出力データと同一になるように、バックアップ制御器からの出力データを処理する。

20

【0015】

階層の下位レイヤの制御器は、上位レイヤの複数の異なる制御器の各々から出力データを受信する。上位レイヤの制御器からの出力データは、複数の異なるトランザクションとして供給される。いくつかの実施形態において、下位レイヤの制御器は、異なる制御器から受信した出力データから自身の出力データを生成し、自身の出力データを単一のトランザクションとして更に下位のレイヤの制御器のセットに送出する。

【0016】

更に詳細な実施形態を以下の節で説明する。詳細には、最初に第Ⅰ節において、論理ネットワーク及び物理ネットワークを制御するいくつかの実施形態のネットワーク制御システムを説明する。次に第Ⅱ節において、更新速度の最低限化を説明する。第Ⅲ節において、本発明のいくつかの実施形態が実現される電子システムを説明する。

30

【0017】

Ⅰ．ネットワーク制御システム

図１は、ネットワーク制御器が被管理転送要素のセットの転送挙動を定義するために被管理転送要素のセットにプッシュする転送状態情報を計算するネットワーク制御システム１００を示す。ネットワーク制御システム１００は、論理制御器１１０と、２つの物理制御器１１５及び１２０と、３つの被管理転送要素１２５～１３５とを含む。ネットワーク制御システム１００は、２つの物理制御器１１５及び１２０が３つの被管理転送要素に状態をプッシュする単純化された例を表す。多くの場合、いくつかの実施形態のネットワーク制御システムは、非常に多くの制御器及び数百個又は数千個の被管理転送要素を含む。

40

【0018】

いくつかの実施形態において、ネットワーク制御器１１０～１２０は、転送状態の計算を実行し、当該状態をフローエントリの形態で被管理転送要素にプッシュする。いくつかの実施形態のネットワーク制御器は、論理ネットワークを定義する論理制御プレーン（ＬＣＰ）データを受信し、当該ＬＣＰデータを物理制御プレーン（ＰＣＰ）データに変換して被管理転送要素１２５～１３５に送出する。いくつかの実施形態において、論理ネットワークの論理制御プレーンは、論理アドレス空間においてエンドマシン（例えば、仮想マシン）を接続する１つ以上の論理転送要素（例えば、論理スイッチ、論理ルータ）を定義

50

する。論理転送要素は、送出元マシンからのパケットが論理空間において宛先マシンに転送される方法（例えば、論理ポートと仮想マシンのMACアドレスとの結び付き）を定義する。更に、いくつかの実施形態において、LCPは、論理転送要素により実現される論理ポリシー（例えば、アクセス制御リスト）を定義する。LCP及びその構造が実現される物理ネットワークにとって、LCP及びその構造は既知でない。

【0019】

いくつかの実施形態のネットワーク制御器は、被管理転送要素にプッシュされるPCPデータに到達するために、LCPデータの複数の異なる変換を実行する。いくつかの実施形態において、制御器はLCPデータを論理転送プレーン（LFP）データに変換し、その後、LFPデータをPCPデータに変換する。LFPデータは、論理空間においてパケットを転送するための転送エントリを定義する。すなわち、LFPデータは、単に論理ポートとアドレスとを結び付けるだけでなく、アドレスが一致する場合にパケットを論理ポートに転送することを示すエントリを含む。

10

【0020】

LFPデータからPCPデータへの変換は、論理転送エントリを物理ネットワークに組み込む。PCPエントリは、物理ネットワーク内の論理アドレス空間における転送（例えば、物理ポートに対する論理ポートのマッピング等）を実行するための情報を含む。

【0021】

いくつかの実施形態において、被管理転送要素にプッシュするPCPの計算は、制御器により形成される階層内の異なる制御器レイヤ間で分散される。例えばいくつかの実施形態において、論理制御器110は、少なくとも1つの論理転送要素を管理する。論理制御器110は、図1の右半分で示すように、LCPからLFPへの変換及びその後のLFPから汎用PCP（UPCP）への変換を実行する。UPCPデータはフローエントリを含む。これらのフローエントリは、何らかの被管理転送要素に特有のデータを含むようにカスタマイズされておらず、特定の物理的実現に特有のデータに対する抽象的概念（例えば、ポート番号、トンネル識別子等）のみを含む。

20

【0022】

いくつかの実施形態において、特定の論理転送要素を管理する論理制御器は、1つ又は複数の物理制御器にUPCPデータを送出する。例えば論理制御器110は、2つの物理制御器115及び120にUPCPデータを送出する。被管理転送要素の各々は、マスタ物理制御器により管理される。従って、複数の被管理転送要素にわたり実現される論理転送要素に対するUPCPデータは、それらの転送要素を管理する複数の異なるマスタ物理制御器に送出されてもよい。示すように、物理制御器115は、2つの被管理転送要素125及び130を管理するマスタ制御器である。物理制御器120は、被管理転送要素135を管理するマスタ制御器である。

30

【0023】

物理制御器又は被管理転送要素と同一の物理マシン内のシャシ制御器（図1では不図示）のいずれかにおいて、UPCPデータはカスタマイズ済PCP（CPCP）データに変換される。CPCPデータは、特定の被管理転送要素に特有のカスタマイズデータが記入された物理制御プレーンデータである。述べられるように、いくつかの実施形態において、物理制御器は、被管理転送要素から受信した情報を使用して当該変換を実行する。他の実施形態において、物理制御器は、被管理転送要素が存在するホストマシンにUPCPデータを送出する通過点として動作し、この場合、制御器の論理（シャシ制御器）がUPCPからCPCPへの変換を実行する。

40

【0024】

被管理転送要素125～135は、ネットワーク制御器により管理される（例えば、ネットワーク制御器から転送状態情報を受信する）ソフトウェア又はハードウェアの転送要素である。いくつかの実施形態において、被管理転送要素は、ホストマシン上（例えば、ホストマシンのユーザ空間及び／又はカーネル内）で動作するソフトウェアの転送要素である。これらの被管理転送要素は、エンドマシン140～160からパケットを受信し、

50

パケットに対して論理処理を実行し、物理ネットワークを介して宛先（例えば、異なる被管理転送要素に同様に接続された別のエンドマシン）にパケットを送出する。

【0025】

エンドマシン140～160は、物理マシン又は仮想マシンであってもよい。いくつかの実施形態において、仮想マシンであるエンドマシンは、それらに対するパケットを転送する被管理転送要素を有する同一ホスト内で動作する。複数の物理ネットワークに属する仮想マシンが単一のホストマシン内に位置してもよい（例えば、エンドマシン140及び145は、被管理転送要素125が位置する同一のホストマシン内に位置してもよい）、被管理転送要素の各々が複数の異なる論理転送要素を実現してもよい。更に、上述のように、単一の論理転送要素は一般に、多くの被管理転送要素にわたり実現される。

10

【0026】

仮想マシンを有するホスト上のネットワークエッジに位置する被管理転送要素に加えて、いくつかの実施形態は第2レベル非エッジ被管理転送要素を更に含む（プールノード又はサービスノードと呼ぶ場合もある）。エッジ被管理転送要素がパケットに対する処理の全てを実行できない場合（例えば、宛先MACアドレスと論理ポートとを結び付けるフローエントリを有さないため）、プールノードがパケットを処理して宛先に送出手のために、エッジ被管理転送要素はプールノードにパケットを送出する。

【0027】

図2は、いくつかの実施形態のネットワーク制御器200のアーキテクチャの例を概念的に示す。ネットワーク制御器200は、それが処理するデータの種類の依存して、論理制御器、物理制御器又はシャシ制御器として機能できる。

20

【0028】

論理制御器である場合、ネットワーク制御器200はLCPデータを入力として受信する。いくつかの実施形態において、ネットワーク制御器200は、LCPデータをLFPデータに変換した後にUPCPデータに変換する。ネットワーク制御器200は、論理制御器であるネットワーク制御器200が管理する論理転送要素を実現する被管理転送要素のマスタである物理制御器のセットにUPCPデータをプッシュする。

【0029】

いくつかの実施形態の物理制御器である場合、ネットワーク制御器200はUPCPデータを入力として受信し、UPCPデータをCPCPデータに変換する。その後、ネットワーク制御器は、ネットワーク制御器200がマスタである被管理転送要素のセットにCPCPデータをプッシュする。他の実施形態において、物理制御器であるネットワーク制御器200は、被管理転送要素のセットが動作するホストにおいて動作するシャシ制御器のセットにUPCPデータを中継する。これらの実施形態において、ネットワーク制御器200は、この被管理転送要素のセットのマスタである。

30

【0030】

シャシ制御器である場合、ネットワーク制御器200は、物理制御器のセットからのUPCPデータを入力として受信する。ネットワーク制御器200は、UPCPデータをシャシ制御器が管理する被管理転送要素に対するCPCPデータに変換し、被管理転送要素にCPCPデータを送出する。

40

【0031】

図2に示すように、ネットワーク制御器200は、規則エンジン入力テーブルのセット210、関数テーブル及び定数テーブルのセット215、インポータ220、規則エンジン225、規則エンジン出力テーブルのセット245、変換器250、エクスポータ255、非一時的トランザクションデータベース（PTD）260及びコンパイラ235を含む。コンパイラ235は、制御器の他の構成要素と異なる時点で動作する制御器の1つの構成要素である。コンパイラは、開発者が特定のネットワーク制御器及び／又は仮想環境に対する規則エンジンを指定する必要がある場合に動作し、制御器の残りのモジュールは、制御器が他の制御器又は被管理転送要素とインタフェースする時に実行時に動作する。

【0032】

50

いくつかの実施形態において、コンパイラ 235 は、宣言型言語で指定される相対的に小さいセット（例えば、数百行）の宣言型命令 240 を受信し、制御器のテーブルマッピングを実行する規則エンジン 225 の動作を指定する大きいセット（例えば、数千行）のコード（すなわち、オブジェクトコード）にこれらを変換する。このように、コンパイラにより、ネットワーク制御器の開発者によるネットワーク制御器の定義及び更新処理が大幅に簡略化される。これは、コンパイラにより、開発者が、ネットワーク制御器の複雑なマッピング動作のコンパクトな定義を可能にする高度なプログラミング言語を使用でき、その後、任意の数の変更（例えば、ネットワーク制御器によりサポートされる論理ネットワーク化機能の変更、ネットワーク制御器の所望の挙動への変更等）に応答して当該マッピング動作を更新できるからである。更に、コンパイラにより、開発者は、マッピング動作を定義する際にイベントがネットワーク制御器に到着する順序を考慮する必要がなくなる。また、開発者は、ネットワーク制御器 200 が論理制御器、物理制御器又はシャーシ制御器として機能するように、異なる規則セットを用いてネットワーク制御器 200 をプログラムする。

10

【0033】

いくつかの実施形態において、規則エンジン（RE）入力テーブル 210 は、ネットワーク制御器 200 がどの種類のネットワーク制御器として動作するかに基づいて、異なる種類のデータを有するテーブルを含む。ネットワーク制御器 200 が論理制御器として動作する場合、入力テーブル 210 は、LFP データにマッピングされる必要のある LCP データを含み且つ UPCP データにマッピングされる必要のある LFP データを含む。ネットワーク制御器 200 が物理制御器又はシャーシ制御器として動作する場合、入力テーブル 210 は、CPCP データにマッピングされる必要のある UPCP データを含む。

20

【0034】

RE 入力テーブル 210 に加えて、ネットワーク制御器 200 は、規則エンジン 225 がテーブルマッピング動作のための入力を収集するために使用する他の種々のテーブル 215 を含む。これらのテーブル 215 は、規則エンジン 225 がテーブルマッピング動作を実行するために必要とする定数に対する規定値を格納する定数テーブルを含む。例えば定数テーブル 215 は、値 0 として定義される定数「zero」、値 4000 として定義される定数「dispatch_port_no」、並びに値 0xFF:FF:FF:FF:FF:FF として定義される定数「broadcast_mac_addr」を含む。

30

【0035】

規則エンジン 225 が定数を参照する場合、定数に対して定義された対応する値が実際は検索され且つ使用される。更に、定数テーブル 215 内の定数に対して定義された値は、変更及び／又は更新されてもよい。このように、定数テーブル 215 は、規則エンジン 225 の動作を指定するコードを書き換えるか又は再度コンパイルする必要なく、規則エンジン 225 が参照する定数に対して定義された値を変更する機能を提供する。テーブル 215 は、出力テーブル 245 をポピュレートするために必要な値の計算に使用するために規則エンジン 225 が必要とする関数を格納する関数テーブルを更に含む。

40

【0036】

規則エンジン 225 は、入力データを出力データに変換する 1 つの方法を指定するテーブルマッピング動作を実行する。規則エンジン（RE）入力テーブルのうちの 1 つが変更される度に規則エンジンはテーブルマッピング動作のセットを実行し、その結果、1 つ以上の RE 出力テーブル内の 1 つ以上のデータタプルが変更される。いくつかの実施形態において、ネットワーク制御システムは、nLog と呼ばれるデータログデータベース言語の変形を使用して規則エンジン 225 を作成する。データログと同様に、nLog が提供する開発者が異なるイベントの発生時に実行される異なる動作を指定できるようにする宣言規則及び演算子の数は少ない。いくつかの実施形態において、nLog は、nLog の動作を高速化するために、データログにより提供される演算子の限られたサブセットを提供する。例えばいくつかの実施形態において、nLog は、いずれかの宣言規則における

50

AND演算子の使用のみを許可する。

【0037】

図2に示すように、規則エンジン225は、イベントプロセッサ222、複数のクエリプラン227及びテーブルプロセッサ230を含む。各クエリプランは、RE入力テーブルのうちの1つに対する変更が発生した時に実行される結合動作のセットを指定する規則セットである。以下、そのような変更を入力テーブルイベントと呼ぶ。各クエリプランは、コンパイラ235により宣言セット240内の1つの宣言規則から生成される。いくつかの実施形態において、2つ以上のクエリプランが1つの宣言規則から生成される。例えばクエリプランは、1つの宣言規則により結合されるテーブルの各々に対して作成される。すなわち、宣言規則が4つのテーブルを結合するように指定する場合、4つの異なるクエリプランがこの1つの宣言から作成される。いくつかの実施形態において、クエリプランは、nLogの宣言的言語を使用することにより定義される。

10

【0038】

規則エンジン225のイベントプロセッサ222は、各入力テーブルイベントの発生を検出する。異なる実施形態のイベントプロセッサは、入力テーブルイベントの発生を異なる方法で検出する。いくつかの実施形態において、イベントプロセッサは、RE入力テーブルのレコードに対する変更の通知に対するRE入力テーブルとのコールバックに登録する。そのような実施形態において、イベントプロセッサ222は、レコードのうちの1つが変更されたという通知をRE入力テーブルから受信した場合に入力テーブルイベントを検出する。

20

【0039】

検出された入力テーブルイベントに応答して、イベントプロセッサ222は、(1)検出されたテーブルイベントに対する適切なクエリプランを選択し、(2)テーブルプロセッサ230にクエリプランを実行するように命令する。いくつかの実施形態において、クエリプランを実行するために、テーブルプロセッサ230は、クエリプランにより指定された結合動作を実行し、1つ以上の入力テーブル210及び種々のテーブル215から1つ以上のデータ値セットを表す1つ以上のレコードを生成する。その後、いくつかの実施形態のテーブルプロセッサ230は、(1)結合動作により生成されたレコードからデータ値サブセットを選択する選択動作を実行し、(2)1つ以上のRE出力テーブル245に選択されたデータ値サブセットを書き込む。

30

【0040】

いくつかの実施形態において、RE出力テーブル245は、論理ネットワーク要素及び物理ネットワーク要素の双方のデータ属性を格納する。テーブル245が規則エンジン225のテーブルマッピング動作の出力を格納する場合、テーブル245をRE出力テーブルと呼ぶ。いくつかの実施形態において、RE出力テーブルは複数の異なるカテゴリでグループ化される。例えばいくつかの実施形態において、これらのテーブルは、RE入力テーブル及び／又は制御器出力テーブルである場合がある。テーブルにおける変更により規則エンジンがクエリプランの実行を必要とする入力イベントを検出する場合、テーブルはRE入力テーブルである。RE出力テーブル245は、規則エンジンに別のクエリプランを実行させるイベントを生成するRE入力テーブル210である場合もある。そのようなイベントを内部入力イベントと呼び、これは、インポータ220により生成されたRE入力テーブルの変更により発生するイベントである外部入力イベントと対比される。

40

【0041】

テーブルにおける変更によりエクスポート255が変更を別の制御器又は被管理転送要素にエクスポートする場合、テーブルは制御器出力テーブルである。RE出力テーブル245内のテーブルは、RE入力テーブルである場合もあり、制御器出力テーブルである場合もあり、あるいはRE入力テーブル及び制御器出力テーブルの双方である場合もある。いくつかの実施形態において、RE入力テーブル及びRE出力テーブルは、関係型データベース管理システム(RDBMS)のテーブルである。これらのテーブルは、ネットワーク制御器の一次データ記憶構造である関係型データベースのデータ構造として格納される

50

。

【0042】

エクスポート255は、RE出力テーブル245の制御器出力テーブルに対する変更を検出する。異なる実施形態のエクスポートは、制御器出力テーブルイベントの発生を異なる方法で検出する。いくつかの実施形態において、エクスポートは、制御器出力テーブルのレコードに対する変更の通知に対する制御器出力テーブルとのコールバックに登録する。そのような実施形態において、エクスポート255は、レコードのうちの1つが変更されたという通知を制御器出力テーブルから受信した場合に出力テーブルイベントを検出する。

【0043】

検出された出力テーブルイベントに応答して、エクスポート255は、変更された制御器出力テーブル内の変更されたデータタプルの一部又は全てを取得し、この変更されたデータタプルを他の制御器又は被管理転送要素に伝搬する。詳細には、ネットワーク制御器200が論理制御器として動作する場合、エクスポート255は、物理制御器との間に確立された通信チャンネル（例えば、遠隔手続き呼出し（RPC）チャンネル）のセットを介して物理制御器のセットにUPCPデータを伝搬する。ネットワーク制御器200が物理制御器として動作する場合、いくつかの実施形態のエクスポート255は、シャーシ制御器との間に確立された通信チャンネルのセットを介してシャーシ制御器のセットにUPCPデータを伝搬する。他の実施形態のエクスポート255は、各被管理転送要素との間に確立された通信チャンネル対（例えば、OpenFlowチャンネル及びコンフィギュレーションチャンネル）を介して被管理転送要素のセットにCPCPデータを伝搬する。ネットワーク制御器200がシャーシ制御器として動作する場合、いくつかの実施形態のエクスポート255は、各被管理転送要素との間の通信チャンネル対（例えば、OpenFlowチャンネル及びコンフィギュレーションチャンネル）を介して被管理転送要素のセットにCPCPデータを伝搬する。

【0044】

いくつかの実施形態において、ネットワーク制御器は、それが管理する責任を有さないデータを出力テーブル245内に保持しない。しかし、そのようなデータは、変換器250によりPTDに格納できる形式に変換され、PTD260に格納される。PTDは、ネットワーク制御器の2次記憶構造である。ネットワーク制御器200のPTDは、1つ以上の他のネットワーク制御器に当該データを伝搬し、それにより、それらのネットワーク制御器の中でデータを管理する責任を有するいくつかのネットワーク制御器がデータを処理できる。

【0045】

いくつかの実施形態において、ネットワーク制御器は、データの回復のために、出力テーブル245に格納されたデータ（すなわち、ネットワーク制御器が管理する責任を有するデータ）をPTDに更に格納する。そのようなデータは、変換器250により同様に変換され、PTDに格納され、他の制御器インスタンスの他のPTDに伝搬される。従って、これらの実施形態において、制御器インスタンスの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データに対する全ての構成データを有する。すなわち、いくつかの実施形態において、各PTDは論理及び物理ネットワークの構成の全体像を含む。

【0046】

インポート220は、多くの異なる入力データ送出元とインタフェースし、入力データを使用して入力テーブル210を変更又は作成する。いくつかの実施形態のインポート220は、ネットワーク制御器200が論理制御器として動作する場合、ユーザ入力（例えば、アプリケーションプログラミングインタフェース（API）呼出しの形態である）をLCPデータに変換する入力変換制御器（不図示）を介してユーザ（テナント）から入力データを受信する。いくつかの実施形態において、インポート220は通信チャンネルを介してLCPデータを受信する。インポート220は、PTDを介して他の制御器インスタンスから受信したデータを入力テーブル210を変更又は作成するための入力データとし

て使用できるように、PTD260と更にインタフェースする。更に、インポータ220は、RE出力テーブル245のRE入力テーブル及び制御器出力テーブルにおける変更も検出する。出力テーブル245において生成及び格納されたLFPデータは、規則エンジン225がUPCPデータを生成するために、インポータ220により規則エンジン225にフィードバックされる。

【0047】

ネットワーク制御器200が物理制御器として動作する場合、インポータ220は、論理制御器のセットとの間に確立された通信チャネルのセットを介して論理制御器のセットからUPCPデータを取得する。ネットワーク制御器200がシャシ制御器として動作する場合、インポータは、物理制御器のセットとの間に確立された通信チャネルのセットを介して物理制御器のセットからUPCPデータを取得する。

10

【0048】

図2ではここまで、入力テーブル210は制御器の階層の上位レイヤの制御器からの入力を含み、出力テーブル245は制御器の階層の下位レイヤの制御器又は被管理転送要素のセットに対する出力を含むと説明した。入力及び出力が逆方向に送受信される場合もある。すなわち、そのような場合、ネットワーク制御器は下位レイヤの制御器又は被管理転送要素から入力を取得し、上位レイヤの制御器に出力を送出する。例えばネットワーク制御器200は、ユーザから送われる要求を受信し、要求を下位レイヤの制御器のセット又は被管理転送要素のセットに分散してもよい。これらの分散された要求は被管理転送要素に到達し、被管理転送要素は応答を作成する。応答は、インポータを介して入力としてネットワーク制御器200に戻る。規則エンジン255は、テーブルマッピング動作を実行して応答を組み合わせ、ネットワーク制御器200に要求を送出した制御器に送出する応答を生成する。図9及び図10を参照して、要求及び応答の処理に関する更なる詳細を後で説明する。

20

【0049】

ネットワーク制御器が階層を形成するネットワーク制御システムを説明したが、以下の第II節では、トランザクションを組み合わせることによりネットワーク制御システムにおけるチャーン化を最小限にすることを説明する。

【0050】

II. 更新速度の最低限化

30

A. 外部入力の並べ替え

ネットワーク制御システムにおいて、ネットワーク制御器は、物理ネットワークにわたり論理ネットワークを実現するためにネットワーク状態を管理する。ネットワーク状態は不変でなく、状態が変化する場合、状態の更新はネットワーク全体にわたり被管理転送要素に分散される必要がある。これらのネットワーク状態の更新は、少なくとも3つの理由から生じる。第1に、論理パイプラインにより実施されるネットワークポリシーが再構成される（例えば、論理ネットワークの管理者によるアクセス制御リストの更新）ため論理ポリシーが変化する場合、ネットワーク状態が変化する。第2に、作業負荷の動作の変化の結果、ネットワーク状態が変化する。例えば仮想マシンを第1のノードから第2のノードに移動する場合、論理的観点は変化しない。しかし、VMが接続する論理ポートの物理的位置が異なるため、この移動によりネットワーク状態が更新される必要がある。第3に、装置の追加、除去、アップグレード及び再構成等の物理的再構成イベントの結果、ネットワーク状態が変化する場合がある。

40

【0051】

一般的なユーザ主導によるポリシー構成の変更では小さい漸進的变化が生じ、転送状態に対する当該漸進的变化は効率的に計算可能であるが、フェイルオーバーの状況によりnLog計算エンジンに対する入力が大きく変更される場合がある。送出元制御器から入力を受信するように構成される受信側制御器であり、送出元制御器が故障し且つ新しい制御器が送出元制御器のタスクを含む場合の受信側制御器を考慮する。新しい制御器はバックアップ制御器であるため、状態を事前に計算されていたが、受信側制御器は依然として古い

50

送出元から新しい送出元へのフェイルオーバーを行う必要がある。

【0052】

いくつかの実施形態において、受信側制御器は、故障した制御器から受信した全ての入力を単純に削除し（入力の影響を取り消し）、古い入力及び新しい入力が完全に同一でなくてもほぼ同一である可能性が非常に高いと予想できる場合でも、新しい制御器からの新しい入力をnLog計算エンジンに供給する。計算のトランザクション性により、新しい送出元が起動し且つ計算が固定点（例えば、計算が所定の入力データに対して行われる時点）に到達する前に転送状態の変更が公開されることは阻止されるが、計算のオーバーヘッドは膨大になる。最初に状態を除去するために計算され、次に状態を再度確立するために計算されるため、転送状態全体が2回計算される。

10

【0053】

いくつかの実施形態において、受信側制御器は、古い送出元からの入力と新しい送出元からの入力との相違点を識別し、変更された入力に対してのみ転送状態の変更を計算する。これにより、オーバーヘッドが完全に解消される。しかし、トランザクションの計算及び固定点到達機能を用いる場合、いくつかの実施形態の受信側制御器は、相違点を識別することなく同一の結果を得ることができる。相違点を識別することなく入力の送出元から別の送出元に徐々に効率的に移行するために、ネットワーク制御システムは単に、古い送出元からの入力を削除することから始めるのではなく、古い送出元からの入力を依然として使用しつつ新しい送出元からの入力を計算エンジンに供給する。その後、ネットワーク制御システムは、新しい送出元が新しい送出元からの入力に対する固定点に到達するまで待ち、新しい送出元がそのような固定点に到達してから、古い送出元からの入力を削除する。

20

【0054】

外部入力／イベントをこのように並べ替えることにより、いくつかの実施形態のnLog計算エンジンは重複を検出でき、古い状態を完全に削除するというオーバーヘッドを回避できる。古い送出元から状態を削除する必要があるため、受信側制御器は新しい送出元が固定点に到達するまでトランザクションをコミットしない。新しい送出元が固定点に到達すると、受信側制御器は、変更された入力による転送状態（すなわち、出力状態）に対する変更を消費側転送要素にプッシュする。変更が著しい場合、この手法により、一時的なメモリの使用が増加するというコストが生じる。いくつかの実施形態において、送出元制御器が固定点に到達した場合、送出元制御器はバリアを送出する。バリアが受信側制御器において受信される場合、受信側制御器は、送出元制御器が固定点に到達したことを認識する。

30

【0055】

図3は、論理制御器310から入力を受信する物理制御器305を概念的に示す。特に、図3は、論理制御器310に障害が発生し、論理制御器335が更新を計算し且つ物理制御器305に送出するタスクを引き継ぐ場合の物理制御器305の入力処理を4つの異なる段階301～304で示す。論理制御器335は、論理制御器310に対するホットスタンバイ論理制御器である。

【0056】

物理制御器305は、制御器200の対応する構成要素と同様であるインポータ315、規則エンジン320、入力テーブル325及び出力テーブル330を含む点で図2を参照して上述したネットワーク制御器200と同様である。説明を簡略にするために、物理制御器305の全ての構成要素を図3に示すわけではない。

40

【0057】

第1の段階301において、論理制御器310は、白色の平行四辺形で示す入力の変更1及び2を物理制御器305に送出している。入力の変更は、制御器の入力テーブルの1つ以上のレコードに対する変更である。いくつかの実施形態において、入力の変更はデータタプルの形態である。論理制御器335も同様に、同一の変更1及び2を物理制御器305に送出する。論理制御器310からの変更1及び2と視覚的に区別するために、バッ

50

クアップ論理制御器 3 3 5 から送出された変更 1 及び 2 を灰色の平行四辺形で示す。

【0058】

第 2 の段階 3 0 2 において、物理制御器 3 0 5 は、論理制御器 3 1 0 から変更 1 及び 2 を受信し且つ論理制御器 3 3 5 から変更 1 及び 2 を受信した。しかし、インポータ 3 1 5 は、論理制御器 3 1 0 からの変更 1 及び 2 のみを用いて入力テーブル 3 2 5 を更新し、バックアップ論理制御器 3 3 5 からの変更 1 及び 2 を記憶構造（不図示）に保持している。

【0059】

いくつかの実施形態において、物理制御器 3 0 5 は、論理制御器 3 3 5 が論理制御器 3 1 0 のバックアップ制御器であることを認識しない。すなわち、物理制御器 3 0 5 から見た場合、論理制御器 3 1 0 及び 3 3 5 は同一の入力の変更を供給する 2 つの制御器である。物理制御器 3 0 5 は、制御器のうちの一方からの変更を使用することをローカルに決定し、変更を使用していた制御器に障害が発生した場合、他方の制御器に切り替える。段階 3 0 2 において、物理制御器 3 0 5 は論理制御器 3 1 0 からの変更を使用する。

【0060】

段階 3 0 2 は、論理制御器 3 1 0 に障害が発生し、論理制御器 3 3 5 が論理制御器 3 1 0 の障害発生後に変更 3 及び 4 を送出していることを更に示す。変更 4 が論理制御器 3 3 5 からのトランザクションの最後の変更であることを示すために、太枠を用いて変更 4 を示す。換言すると、変更 3 及び 4 はトランザクションを形成し、変更 4（又は変更 4 の後の別個のデータ）は、1 つのトランザクションに対する入力セットの末尾を示すバリアを有する。規則エンジン 3 2 0 は、例えば他の変更（不図示）の処理がまだ終了していないため、変更 1 及び 2 をまだ処理していない。

【0061】

第 3 の段階 3 0 3 は、規則エンジン 3 2 0 がテーブルマッピング動作を実行して、変更 1 及び 2 から出力の変更のセットを生成したことを示す。出力の変更は、入力の変更により変更される入力テーブルにおいてテーブルマッピング動作を実行した結果、制御器の出力テーブルの 1 つ以上のレコードに対して行われる変更である。いくつかの実施形態において、出力の変更はデータタブルの形態である。これらの出力の変更が論理制御器 3 1 0 からの変更 1 及び 2 を処理した結果であることを示すために、変更 1 及び 2 を含む破線の枠として出力の変更を示す。また、第 3 の段階 3 0 3 において、インポータ 3 1 5 は、記憶構造に保持された論理制御器 3 3 5 からの変更 1 及び 2 を用いて入力テーブル 3 2 5 を更新した。また、論理制御器 3 1 0 に障害が発生し、論理制御器 3 3 5 が論理制御器 3 3 5 に切り替えられ且つ変更が論理制御器 3 3 5 から受信されるため、インポータ 3 1 5 は論理制御器 3 1 0 からの変更 1 及び 2 を除去した。更に、物理制御器 3 0 5 は論理制御器 3 3 5 から変更 3 及び 4 を受信した。インポータ 3 1 5 は、変更 3 及び 4 を用いて入力テーブル 3 2 5 を更新する。

【0062】

第 4 の段階 3 0 4 は、規則エンジン 3 2 0 がテーブルマッピング動作を実行し、バックアップ論理制御器 3 3 5 を介して受信した変更 1 ～ 4 から出力の変更を生成したことを示す。変更 1 ～ 4 を含む破線の枠として示す出力の変更は、インポータが変更 1 及び 2 を用いて入力テーブルを 2 回更新しなかった（1 回目は論理制御器 3 1 0 からの変更 1 及び 2 を用い、2 回目は論理制御器 3 3 5 からの変更 1 及び 2 を用いる）場合に生成される出力の変更と同一であることを示す。これは、いくつかの実施形態の規則エンジンが、重複する入力の変更に対してテーブルマッピング動作を実行することにより重複する出力の変更を生成しないからである。

【0063】

物理制御器が上位レイヤの制御器からのトランザクションを形成する全ての入力の変更を処理したため、論理制御器 3 3 5 は自身の固定点に到達した。その後、物理制御器 3 3 5 は、被管理転送要素のセット又はシャース制御器のセットにこの出力の変更のセットを送出する。図 3 は、物理制御器による論理制御器のフェイルオーバの処理を示す。しかし、シャース制御器が物理制御器のフェイルオーバを同様に処理してもよいことが当業者に

10

20

30

40

50

は認識されるだろう。

【0064】

図4は、ネットワーク制御器の階層における上位レイヤの送出元制御器のフェイルオーバーを処理するためにいくつかの実施形態が実行する処理400を概念的に示す。処理400は、入力の変更を生成する2つ以上の送出元制御器から入力の変更を受信する受信側制御器により実行される。いくつかの実施形態において、受信側制御器は、UPCPデータを含む入力の変更を生成する論理制御器のセットから入力の変更を受信する物理制御器である。また、受信側制御器は、UPCPデータを中継する物理制御器のセットから入力の変更を受信するシャシ制御器であってもよい。いくつかの実施形態の受信側制御器は、図3を参照して上述した物理制御器305と同様である。

10

【0065】

処理400は、マスタ送出元制御器及びバックアップ送出元制御器から入力の変更を受信する(405)ことにより開始する。バックアップ制御器は、マスタ制御器が送出するのと同じの入力の変更を受信側制御器に送出するスタンバイ又は冗長制御器である。いくつかの実施形態において、受信側制御器は、2つの送出元制御器のうちのどちらがマスタ制御器であるかを認識しない。受信側制御器は、それらのうちの一方を選択し、選択された送出元制御器からの入力の変更を使用して受信側制御器自身の出力の変更を生成する。説明の便宜上、マスタ送出元制御器は受信側制御器により最初に選択された制御器である。

【0066】

20

次に、処理400において、マスタ制御器からの入力のみを使用して出力の変更を計算する(410)。いくつかの実施形態の処理400において、トランザクション(例えば、バリア間の入力の変更のセット)がマスタ送出元制御器から完全に受信されるまで、バックアップ制御器からの重複する入力の変更を記憶構造に保持する。いくつかの実施形態の処理400において、記憶構造に保持される入力の変更は使用されない。いくつかの実施形態の処理400において、マスタ制御器から受信される入力の変更は入力テーブルから除去されない。

【0067】

処理400において、マスタ送出元制御器に障害が発生したかを判定する(415)。いくつかの実施形態において、送出元制御器は、自身の状態又はハートビートを定期的に送信し、受信側制御器は、状態を使用して送出元制御器が動作しているかを判定する。いくつかの実施形態において、受信側制御器は、送出元制御器が動作しているかを判定するために送出元制御器をポーリングする。処理400において、マスタ送出元制御器に障害が発生したと判定される場合(415)、処理400は430へ進む。430を後で説明する。

30

【0068】

処理400において、マスタ送出元制御器に障害が発生していないと判定される場合(415)、マスタ制御器からバリアを受信したかが判定される(420)。すなわち、受信済みの入力の変更が完全なトランザクションを形成するかが判定される。処理400において、マスタ送出元制御器から完全なトランザクションを受信していないと判定される場合(420)、処理400は405に戻り、マスタ送出元制御器及びバックアップ送出元制御器からの入力の変更の受信を続行する。

40

【0069】

処理400において、マスタ送出元制御器からバリアを受信したと判定される場合(420)、処理400が自身の固定点に到達したかが判定される(425)。いくつかの実施形態の処理400において、マスタ送出元制御器から受信したトランザクションの入力の変更の全ての処理が終了して出力の変更を生成した場合、処理400は固定点に到達したと判定される。その場合、処理400は450へ進む。450を後で説明する。

【0070】

処理400において、マスタ送出元制御器に障害が発生したと判定される場合(415

50

）、バックアップ送出元制御器に切り替えてバックアップ制御器から入力の変更を受信する（４３０）。その後、処理４００において、バックアップ制御器から受信した入力に基づいて出力の変更を計算する（４３５）。いくつかの実施形態では、処理４００において、保持された（４１０）変更を更に使用して出力の変更を計算する。保持された変更は、出力の変更を生成するために使用されたマスタ送出元制御器からの入力の変更の重複する変更である。しかし、処理４００において、マスタ送出元制御器から受信した同一の入力の変更を処理することにより生成された出力の変更は削除されない。処理４００において、保持された重複する入力の変更が依然として処理されるが、処理４００を実行する受信側制御器の規則エンジンは、重複する入力の変更を処理することにより重複する出力の変更を生成しない。いくつかの実施形態の処理４００において、バックアップ送出元制御器に切り替える際に、障害が発生した制御器から受信される変更が入力テーブルから除去される。

10

【００７１】

次に、処理４００において、バックアップ送出元制御器からバリアを受信したが判定される（４４０）。すなわち、受信済みの入力の変更が完全なトランザクションを形成するかが判定される。完全なトランザクションを形成する入力の変更は、保持されている重複する入力の変更とマスタ制御器の障害発生後にバックアップ制御器から受信した入力の変更とを含む。

【００７２】

処理４００において、バリアを受信していないと判定される場合（４４０）、処理４００は４３０に戻り、バックアップ送出元制御器からの入力の変更の受信を続行する。処理４００において、バリアを受信したと判定される場合（４４０）、処理４００が自身の固定点に到達したかが判定される（４２５）。

20

【００７３】

次に、処理４００において、制御器の階層における下位レイヤの制御器のセット又は出力の変更に基づいてデータを転送する被管理転送要素のセットに計算された出力の変更を送出する（４５０）。いくつかの実施形態の処理４００において、出力の変更の末尾にバリアを挿入するか、あるいは出力の変更のうちの最後の変更に完全なトランザクションを示す情報を追加する。その後、処理は終了する。

【００７４】

B．階層型の転送状態計算におけるトランザクション

いくつかの実施形態において、ネットワーク制御器は、複数の制御器から受信側トランザクションの更新を受信する転送要素に更新を供給するネットワーク制御器の２つ以上のレイヤを有する階層を形成する。これらの実施形態において、最上位の制御器は更新をトランザクションで計算するが、下位レイヤの制御器は複数の最上位の制御器から更新を受信してもよく、同様に、転送要素は複数の第２のレベルの制御器から更新を受信してもよい。

30

【００７５】

トランザクションは、境界で変更されずに下方に進む。すなわち、第２のレベルの制御器において処理された最上位のトランザクションは、最上位の制御器から受信したトランザクションの結果として得られた変更のみを含んで転送要素に供給されるトランザクションになる。しかし、トランザクションが転送要素に向かう途中で集約される場合でも、ポリシーの一貫性は維持される。いくつかの実施形態において、第２のレベルの制御器は、受信した複数のトランザクション（異なる最上位の制御器から受信される場合がある）を集約して、転送要素に供給される単一のトランザクションにする。適切な集約レベルを判定するのは（そのようなレベルが存在する場合）、ローカルな決定である。例えばシステムは、トランザクションがデフォルトでは全く集約されないが、キューのトランザクションの数が増加する過負荷状態において、互いに相殺し合う重複する変更を有するトランザクション（同一の送出元からの）を期待してトランザクションを集約する手法を実施してもよい。更に広いネットワークの場合、当該手法はルートフラップダンピングの一種であ

40

50

ると考えられる。

【0076】

図5は、論理制御器510から入力を受信する物理制御器505を概念的に示す。特に、図5は、物理制御器505に入力の変更を供給する論理制御器510からの複数の完全なトランザクションを形成する入力の変更を物理制御器505が集約する動作を4つの異なる段階501～504で示す。物理制御器505は、制御器200の対応する構成要素と同様であるインポータ515、規則エンジン520、入力テーブル525及び出力テーブル530を含む点で図2を参照して上述したネットワーク制御器200と同様である。説明を簡略にするために、物理制御器505の全ての構成要素を図5に示すわけではない。

10

【0077】

第1の段階501において、論理制御器510は物理制御器505に入力の変更1～3を送出している。変更1～3が完全なトランザクションを形成することを示すために、太枠を用いて変更3を示す。すなわち、変更3はバリアを含むか又はバリアを伴う。段階501において、物理制御器505が前にトランザクションの出力の変更のセットを計算して送出したため、入力テーブル525及び出力テーブル530は空である。

【0078】

第2の段階502において、物理制御器505は論理制御器510から変更1～3を受信した。インポータ515は、変更1～3を用いて入力テーブル525を更新した。第2の段階502は、論理制御器がトランザクションを形成する次の入力の変更4及び5のセットを送出していることを更に示す。

20

【0079】

第3の段階503において、物理制御器505は、トランザクションを形成する次の入力の変更4及び5のセットを論理制御器530から受信した。インポータ515は、変更4及び5を用いて入力テーブル525を更新する。第3の段階503は、規則エンジン520がテーブルマッピング動作を実行して、前の段階502において入力テーブル525に入力された変更1～3から出力の変更のセットを生成したことを更に示す。これらの出力の変更が変更1～3を処理した結果であることを示すために、変更1～3を含む破線の枠として出力の変更を示す。

【0080】

また、段階503において、物理制御器505は、(1)制御器が完全なトランザクションを形成する入力の変更のセットを処理することにより出力の変更を生成したため、出力テーブル530内に現在存在する出力の変更を送出するか、あるいは(2)更なる入力の変更が供給されるのを待つかを判定する。いくつかの実施形態において、物理制御器は、特定の基準に基づいて当該判定を行う。例えば物理制御器は、最後の出力の変更のセットを送出してから又は最後のトランザクションを受信してからある期間が経過していない場合、更なる入力の変更が供給されるのを待つ。これらの実施形態のうちのいくつかにおいて、ある期間が経過した場合、物理制御器505は、完全なトランザクションを形成する入力の変更の全てを集約して、トランザクションの出力の変更の単一セットを生成する。

30

40

【0081】

その代わりに又はそれに加えて、いくつかの実施形態の物理制御器505は、入力テーブル525内に有するデータ量を考慮する。これらの実施形態のうちのいくつかにおいて、入力テーブル525が閾値量を上回るデータを有する場合、物理制御器505は、完全なトランザクションを形成する入力の変更の全てを集約して、トランザクションの出力の変更の単一セットを生成する。データ量を考慮する代わりに又はそれに加えて、いくつかの実施形態の物理制御器505は、入力テーブル525が有する完全なトランザクションの数を考慮する。いくつかのそのような実施形態において、入力テーブル525が閾値数を上回る完全なトランザクションを有する場合、物理制御器は完全なトランザクションを形成する入力の変更を集約して、トランザクションの出力の変更の単一セットを生成する

50

。

【0082】

第4の段階504において、物理制御器505は、トランザクションの出力の変更の単一セットを生成するために更なるトランザクションを使用する必要があると判定した。従って、物理制御器505は、変更1～3から計算された出力の変更を送出していない。規則エンジン520は、変更4及び5に対してテーブルマッピング動作を実行して、出力の変更を生成した。示すように、変更4及び5から生成された出力の変更は、変更1～3から生成された出力の変更と共にグループ化される。その後、物理制御器505は、この出力の変更のグループを被管理転送要素のセット又はシャーン制御器のセットに送出する。

【0083】

トランザクションの出力の変更の単一セットは、別の制御器又は被管理転送要素に送出されるトランザクションを形成する。トランザクションは、受信側被管理転送要素の転送状態に適用される変更セットを含む。従って、いくつかの実施形態の制御器は、入力側で複数のトランザクションを集約して出力側に送出する単一のトランザクションを生成することにより変更セットを組み合わせ、それにより、それらの変更の全てが被管理転送要素に共に適用される。

【0084】

図6は、複数の論理制御器635～645から入力の変更を受信する物理制御器610を概念的に示す。特に、図6は、物理制御器610が複数の異なる論理制御器からの複数のトランザクションを形成する入力の変更を集約してトランザクションの出力の変更の単一セットにする動作を5つの段階601～605で示す。物理制御器610は、制御器200の対応する構成要素と同様であるインポータ615、規則エンジン620、入力テーブル625及び出力テーブル630を含む点で図2を参照して上述したネットワーク制御器200と同様である。説明を簡略にするために、物理制御器610の全ての構成要素を図6に示すわけではない。

【0085】

第1の段階601において、論理制御器635は、物理制御器610に入力の変更1を送出している。論理制御器640は、完全なトランザクションを形成する入力の変更2～4を送出している。段階601において、物理制御器610が前にトランザクションの出力の変更のセットを計算して送出したため、入力テーブル625及び出力テーブル630は空である。

【0086】

第2の段階602において、物理制御器610は、論理制御器635及び640から変更1～4を受信した。インポータ615は、変更1～4を用いて入力テーブル625を更新した。第2の段階602は、論理制御器645が完全なトランザクションを形成する変更5及び6を送出していることを更に示す。

【0087】

第3の段階603において、物理制御器610は、トランザクションを形成する入力の変更5及び6を論理制御器645から受信した。インポータ615は、変更5及び6を用いて入力テーブル625を更新する。この時点で、入力テーブル625は変更1～6を有する。第3の段階603は、規則エンジン620がテーブルマッピング動作を実行して、前の段階602で入力テーブル625に入力された変更1～4から出力の変更を生成したことを更に示す。出力の変更の2つのセットが生成された。示すように、第1のセットは、変更1を処理することにより生成された出力の変更を含む。第2のセットは、変更2～4を処理することにより生成された出力の変更を含む。

【0088】

また、段階603において、物理制御器610は、(1)物理制御器が完全なトランザクションを形成する全ての入力の変更を処理することにより出力の変更を生成したため、変更2～4を処理することにより生成されたこれらの出力の変更を送出するか、あるいは(2)更なる入力の変更が供給されるのを待つかを判定する。いくつかの実施形態におい

10

20

30

40

50

て、物理制御器は、図5を参照して上述したように、特定の基準、すなわち、トランザクションの出力の変更のセットを送出してから又は完全なトランザクションを受信してから経過した期間、入力テーブル625内のデータ量、並びに／あるいは入力テーブル625内の完全なトランザクションの数に基づいて当該判定を行う。

【0089】

第4の段階604において、物理制御器610は、トランザクションの出力の変更の単一セットを生成するために更なるトランザクションを使用する必要があると判定した。従って、物理制御器610は、入力の変更2～4から計算された出力の変更を送出していない。規則エンジン620は、変更5及び6に対してテーブルマッピング動作を実行して、対応する出力の変更を生成した。しかし、示すように、この場合、入力の変更5及び6から計算された出力の変更は、入力の変更2～4から計算された出力の変更と共にグループ化される。従って、物理制御器610は、2つのトランザクションを形成する入力の変更2～4のセットと入力の変更5及び6のセットとを処理することで得られた出力を集約することにより、出力の変更の単一セットを生成した。

10

【0090】

第5の段階605において、物理制御器610は、シャース制御器のセット又は被管理転送要素のセットに変更2～6から計算された出力の変更を送出した。物理制御器は、送出された出力の変更を出力テーブル630から除去した。物理制御器は、入力テーブル625から入力の変更2～6を更に除去した。段階605は、入力の変更1が入力テーブル625に残存し且つ入力の変更1から計算された出力の変更が出力テーブル630に残存することを示す。これは、入力の変更1が完全なトランザクションを形成せず、すなわち、変更1を含む完全なトランザクションが物理制御器610において受信されたことを示すバリアを物理制御器が受信していないためである。

20

【0091】

図7は、複数のトランザクションを形成する入力の変更からトランザクションの出力の変更のセットを生成するためにいくつかの実施形態が実行する処理700を概念的に示す。処理700は、入力の変更を生成する2つ以上の送出元制御器から入力の変更を受信する受信側制御器により実行される。いくつかの実施形態において、受信側制御器は、UPCPデータを含む入力の変更を生成する論理制御器のセットから入力の変更を受信する物理制御器である。また、受信側制御器は、UPCPデータを中継する物理制御器のセットから入力の変更を受信するシャース制御器であってもよい。受信側制御器は、図5及び図6の物理制御器505及び605と同様である。

30

【0092】

処理700は、送出元制御器から入力の変更を受信する(705)ことにより開始する。いくつかの実施形態において、異なる送出元制御器からの入力の変更は、異なる論理転送要素のセットに関する。処理700において、それまでに受信した入力の変更を使用して出力の変更を計算する(710)。

【0093】

次に、処理700において、送出元制御器から少なくとも1つの完全なトランザクションを受信したかを判定する(715)。上述のように、完全なトランザクションは、1つの送出元制御器からバリアを受信してから別のバリアを受信するまでの間に当該送出元制御器から受信した入力の変更を含む。処理700において、少なくとも1つの完全なトランザクションを受信していないと判定される場合(715)、処理700は705に戻り、送出元制御器から更なる入力の変更を受信する。

40

【0094】

処理700において、少なくとも1つの完全なトランザクションを受信したと判定される場合(715)、処理700は720に進み、特定の集約基準が満たされるかを判定する。異なる実施形態は、異なる集約基準を有する。例えばいくつかの実施形態において、特定の基準は、最後の出力の変更のセットを送出してから又は最後の完全なトランザクションを受信してから経過した期間を含む。この期間が経過した場合、特定の基準は満たさ

50

れる。その代わりに又はそれに加えて、いくつかの実施形態において、特定の集約基準は、（処理 700 を実行する受信側制御器の）入力テーブル内に有するデータ量を含む。これらの実施形態において、入力テーブルが閾値量を上回るデータを有する場合、特定の基準は満たされる。これらの実施形態のうちのいくつかにおいて、データ量の代わりに又はそれに加えて、特定の基準は、入力テーブルが有する完全なトランザクションの数を含む。これらの実施形態において、入力テーブルが閾値数を上回る完全なトランザクションを有する場合、特定の基準は満たされる。

【0095】

次に、処理 700 において、完全なトランザクションの全てを構成する入力の変更から計算された出力の変更を集約する（725）。いくつかの実施形態では、処理 700 において、完全なトランザクションを形成しない入力の変更から計算される出力の変更は除外される。換言すると、これらの除外される出力の変更は、バリアが受信されていない入力の変更から計算される。

10

【0096】

処理 700 において、制御器の階層における下位レイヤに存在する制御器のセット又は出力の変更に基づいてデータを転送する被管理転送要素のセットに集約された出力の変更を送出する（730）。いくつかの実施形態の処理 700 において、出力の変更の末尾にバリアを挿入するか又は出力の変更のうちの最後の変更に完全なトランザクションを示す情報を追加する。また、送出了た出力の変更を受信側制御器の出力テーブルから除去し、完全なトランザクションを形成し且つ送出了た出力の変更を計算する際に使用した入力の変更を受信側制御器の入力テーブルから除去する。その後、処理は終了する。

20

【0097】

C. ユースケース

1. API

アプリケーションプログラミングインタフェース（API）呼出しの形態である論理転送要素を定義する入力は、API をサポートする入力変換制御器に送出的れる。いくつかの実施形態のネットワーク制御システムは、API の更新をアトミックにレンダリングする。すなわち、構成の変更は、システムを古い状態から新しい状態にアトミックに移行する。詳細には、API 呼出しを受信後、システム内の API 受信コードは nLog エンジンに対する状態を更新し、全ての更新を供給後、システム内の API 受信コードは固定点（計算を収束させる）に到達するのを待ち、nLog に対する変更を行うことにより終了されるトランザクションを信号伝送する。この後、転送状態の更新は、全てが単一のトランザクションの更新でクラスタ階層において下位の制御器に送出的れるか又は転送要素に向けて送出的れる。更新は、受信側要素によりトランザクションで適用される。

30

【0098】

いくつかの実施形態において、API の更新は、更新が単一のトランザクションの更新として受信機に到着する限り、分散格納システム（例えば、制御器内の PTD）にわたり送信される。すなわち、更新が単一のトランザクションの更新として記憶装置に書き込まれ且つ nLog 処理制御器が更新を単一のトランザクションとして受信する限り、状態の更新をプッシュする処理が上述のように継続するため、nLog 計算処理に対する更新を単一のトランザクションの更新として書き込むことができる。

40

【0099】

2. 制御器のフェイルオーバー

論理転送要素のセットを管理するマスタ論理制御器を考慮する。いくつかの実施形態において、制御器は、同一の状態を計算し且つマスタと同様の方法で当該状態をプッシュするホットバックアップを有する。マスタとホットバックアップとの間の 1 つの相違点は、フェイルオーバーが開始するまでバックアップからのストリームが無視されることである。マスタに障害が発生すると、受信側制御器／転送要素は、以下のように古い状態から新しい状態に徐々に移行することによりバックアップに切り替える。

【0100】

50

古いマスタからの状態の更新のストリームを除去／遮断し且つアクティブな更新のストリームが上位の制御器から受信される状態に向けて計算を収束させる代わりに、単に新しいマスタに切り替え、計算を収束させ、古いストリームと新しいストリームとを効果的に合成する。すなわち、これは、双方の送出元が同一又はほぼ同一のストリームを生成するという仮定に基づく。この動作を実行後、制御器は、固定点に到達するのを待つことにより計算が収束するのを待ち、固定点に到達してから古いストリームを完全に除去する。この場合も、固定点に到達するのを待つことにより、制御器は新しい送出元のみを使用して計算を収束させる。この後、制御器は、トランザクションをコミットすることにより古い送出元から新しい送出元への移行を完成させる。これにより、状態の更新が処理される必要があるという信号としてバリアを下位の制御器／転送要素から効果的に渡すための `nLog` の実行時間が信号伝送される。

10

【0101】

D. オンデマンド要求の処理

A P I 要求の処理は、`nLog` エンジンを使用して実現される場合がある。その場合、要求は、`nLog` による A P I 応答の計算をトリガするタプルのセットに変換されて `nLog` エンジンに供給される。A P I 応答は、同様にタプルとして表される。タプルである要求及び応答が要求タプル及び応答タプルと 1 対 1 のマッピングを有するため、応答を待つことは容易である。A P I 要求の処理は、要求とマッチングする応答が到着するのを単に待つことである。要求とマッチングする応答が到着すると、応答に対する計算を実行できる状態になる。

20

【0102】

しかし、要求／応答が 1 対 1 のマッピングを有さない場合、要求の処理が完了したことを認識するのは困難である。その場合、A P I 要求の処理は、要求を供給後に計算の固定点を尋ねてもよい。固定点に到達すると、要求は、生成された全ての応答を有する。要求タプル及び応答タプルが何らかの共通する識別子を有する限り、応答タプルの数に関係なく、応答タプルを識別することは容易である。従って、このユースケースはそのようなコミットの使用を必要としないが、計算を許可するプリミティブは、固定点を待つことである。

【0103】

図 8 は、ネットワーク制御器がユーザからの要求を被管理転送要素に分散し且つ要求に対する応答をユーザに返すネットワーク制御システム 800 を示す。ネットワーク制御システム 800 は、ネットワーク制御システム 800 内の制御器が被管理転送要素の転送挙動を定義するために被管理転送要素にプッシュする転送状態情報を同様に計算する点で図 1 のネットワーク制御システム 100 と同様である。ネットワーク制御システム 800 は、入力変換制御器 805、論理制御器 810、2 つの物理制御器 815 及び 820、並びに 3 つの被管理転送要素 825～835 を含む。ネットワーク制御システム 800 は、2 つの物理制御器 815 及び 820 が要求を 3 つの被管理転送要素に分散する単純化された例を表す。多くの場合、いくつかの実施形態のネットワーク制御システムは、非常に多くの制御器と数百個又は数千個の被管理転送要素とを含む。

30

【0104】

いくつかの実施形態の入力変換制御器 805 は、ユーザから入力を受信する。これらの入力は論理ネットワークの指定を含み、入力変換制御器は、これを論理制御器が後に処理する L C P データに変換する。更に、入力は、論理ネットワークに関する情報に対する要求を更に含んでもよい。例えばユーザからの要求は、統計情報（例えば、ある期間の論理転送要素の論理ポートに対するトラフィック量）を求めてもよい。入力変換制御器 805 は、論理制御器が後に処理するデータタプルの形態の論理要求に要求を変換する。

40

【0105】

いくつかの実施形態において、入力変換制御器 805 は、A P I 呼出しの形態でユーザから入力を受信する。入力変換制御器 805 は A P I をサポートし、ネットワーク管理アプリケーション（例えば、ウェブアプリケーション又はコマンドラインインタフェース（

50

C L I) アプリケーション) は A P I の上に構築可能である。ユーザは、ネットワークアプリケーションを使用して、入力変換制御器に対する入力を取得する。

【0106】

いくつかの実施形態において、ネットワーク制御器 810～820 は、要求の変換を実行し、データタプルの形態で要求を被管理転送要素に分散する。いくつかの実施形態のネットワーク制御器は、要求を被管理転送要素に分散する前に複数の異なる要求変換を実行する。詳細には、論理制御器 810 は、入力変換制御器 805 から論理要求を受信する。いくつかの実施形態において、論理要求は、論理ネットワークの論理属性を用いて指定される。論理要求の一例は、特定の論理転送要素の特定の論理ポートに関する情報に対する要求である。この要求は、論理ポートの名称又はアドレス、並びに論理転送要素の名称又はアドレスを用いて作成される。

10

【0107】

論理制御器 810 は、この論理要求を汎用要求に変換する。いくつかの実施形態において、汎用要求は、論理ネットワークを実現する被管理転送要素の属性を用いて指定される。しかし、これらの属性は、特定の物理的実現に特有ではない抽象的な用語で表される（例えば、ポート番号、トンネル識別子等）。例えば汎用要求は、物理ポートの実際のポート番号を使用する代わりに、いずれかの被管理転送要素の物理ポートの名称を使用して作成される。

【0108】

いくつかの実施形態において、論理制御器 810 は、任意の数の物理制御器に当該汎用要求を送出する。例えば論理制御器 810 は、2つの物理制御器 815 及び 820 に汎用要求を送出する。いくつかの実施形態において、汎用要求は、要求を識別するための識別子を有する。この識別子は、要求と対応する応答とをマッチングさせるために使用される。応答を後で説明する。

20

【0109】

各被管理転送要素は、マスタ物理制御器により管理される。従って、複数の被管理転送要素にわたり実現される論理転送要素に対する論理要求は、これらの転送要素を管理する複数の異なるマスタ物理制御器に送出されてもよい。示すように、物理制御器 815 は、2つの被管理転送要素 825 及び 830 を管理するマスタ制御器である。物理制御器 820 は、被管理転送要素 835 を管理するマスタ制御器である。

30

【0110】

物理制御器又は被管理転送要素と同一の物理マシン内のシャーシ制御器（図 8 に不図示）のいずれかにおいて、汎用要求はカスタマイズ済要求に変換される。いくつかの実施形態において、カスタマイズ済要求は、被管理転送要素に特有の被管理転送要素の属性を用いて指定される。例えば被管理転送要素に対するカスタマイズ済要求は、被管理転送要素の物理ポートに対してローカルに使用される実際のポート番号で作成される。物理制御器がシャーシ制御器に U P C P データを送出する通過点である実施形態において、物理制御器はシャーシ制御器に汎用要求を送出する通過点である。

【0111】

被管理転送要素 825～835 は、図 1 の被管理転送要素 125～135 と同様である。エンドマシン 840～860 は、図 1 のエンドマシン 140～160 と同様である。被管理転送要素 825～835 は、カスタマイズ済要求が問い合わせる情報を収集する。被管理転送要素 825～835 の各々は、カスタマイズ済要求の受信にตอบสนองして、収集した情報を含むカスタマイズ済応答を生成する。

40

【0112】

被管理転送要素は、被管理転送要素がカスタマイズ済要求を受信した物理制御器（又はシャーシ制御器）にカスタマイズ済応答を渡す。物理制御器又はシャーシ制御器のいずれかにおいて、カスタマイズ済応答は必要に応じて集約され、汎用応答に変換される。その後、汎用応答は、物理制御器が汎用要求を受信した論理制御器に渡される。

【0113】

50

例えば物理制御器 8 1 5 は、被管理転送要素 8 2 5 及び 8 3 0 からカスタマイズ済応答を受信し、カスタマイズ済応答を集約して汎用応答に変換する。いくつかの実施形態において、物理制御器 8 2 0 はカスタマイズ済応答を集約する必要がない。物理制御器 8 2 0 は、被管理転送要素 8 2 5 から受信したカスタマイズ済応答の変換のみを行い、論理制御器 8 1 0 に汎用応答を渡す。

【0 1 1 4】

論理制御器 8 1 0 は、論理制御器 8 1 0 が汎用要求を送出した物理制御器から汎用応答を受信する。論理制御器 8 1 0 は汎用応答を集約し、集約された汎用応答を論理応答に変換し、入力変換制御器 8 0 5 に論理応答を渡す。いくつかの実施形態において、入力変換制御器 8 0 5 は、ユーザが管理アプリケーションを介して閲覧する出力に論理応答を変換する。

10

【0 1 1 5】

いくつかの実施形態において、カスタマイズ済応答、汎用応答及び論理応答は、カスタマイズ済要求、汎用要求及び論理要求を指定するために使用されたのと同じ属性でそれぞれ指定される。いくつかの実施形態において、これらの要求及び応答はデータタプルの形態である。

【0 1 1 6】

尚、制御器の階層における制御器は、階層において下位の制御器から複数の応答を受信しない場合がある。例えば要求が特定の被管理転送要素の特定の物理ポートにマッピングされる特定の論理ポートの情報を取得するためのものである場合、論理制御器は、汎用要求を 2 つ以上の物理制御器に分散する必要がないため、物理制御器から 1 つの汎用応答を取得する。

20

【0 1 1 7】

制御器がそれに要求を送出した制御器の階層において上位の別の制御器に応答を渡す場合、制御器は応答をトランザクションで送出する。図 9 は、物理制御器 9 3 5、9 4 0 及び 9 4 5 のセットから受信した汎用応答を集約するいくつかの実施形態の論理制御器 9 1 0 を示す。特に、図 9 は、論理制御器 9 1 0 が複数の異なる物理制御器からの複数のトランザクションを形成する入力の変更を処理してトランザクションの出力の変更の単一セットにすることにより出力の変更を集約する動作を 5 つの段階 9 0 1 ~ 9 0 5 で示す。この場合、論理制御器 9 1 0 は、集約された出力の変更を入力変換制御器（不図示）に渡す。集約された出力の変更は、論理制御器 9 1 0 が入力変換制御器から受信した論理要求により問い合わせられた情報を含む論理応答を含む。

30

【0 1 1 8】

論理制御器 9 1 0 は、制御器 2 0 0 の対応する構成要素と同様であるインポータ 9 1 5、規則エンジン 9 2 0、入力テーブル 9 2 5 及び出力テーブル 9 3 0 を含む点で図 2 を参照して上述したネットワーク制御器 2 0 0 と同等である。説明を簡略にするために、論理制御器 9 1 0 の全ての構成要素を図 9 に示すわけではない。

【0 1 1 9】

第 1 の段階 9 0 1 において、物理制御器 9 3 5 は、論理制御器 9 1 0 に入力の変更 1 ~ 3 を送出している。入力の変更 1 ~ 3 は、論理制御器 9 1 0 から汎用要求を受信したのに応答して物理制御器 9 3 5 により作成される汎用応答を含む。いくつかの実施形態において、入力の変更 1 ~ 3 は汎用要求の識別子を含む。論理制御器 9 1 0 は、識別子を使用して応答と要求とをマッチングする。

40

【0 1 2 0】

いくつかの実施形態において、物理制御器 9 3 5 は、（１）物理制御器 9 3 5 が被管理転送要素のセットから受信するカスタマイズ済応答のセットを集約し且つ（２）集約されたカスタマイズ済応答を汎用応答に変換することにより、汎用応答を作成する。他の実施形態において、物理制御器 9 3 5 は、物理制御器 9 3 5 がシャーシ制御器（不図示）のセットから受信する汎用応答のセットを集約することにより、汎用応答を作成する。シャーシ制御器は、シャーシ制御器が動作するのと同じホストにおいて動作する被管理転送要

50

素のセットからのカスタマイズ済応答のセットを集約することにより、物理制御器 9 3 5 に渡す汎用応答を作成する。

【0 1 2 1】

段階 9 0 1 において、入力テーブル 9 2 5 及び出力テーブル 9 3 0 は、転送状態、要求及び／又は応答に対するレコードを含んでもよい。説明を簡略にするために、これらの他のレコードを図 9 に示さない。

【0 1 2 2】

第 2 の段階 9 0 2 において、論理制御器 9 1 0 は、物理制御器 9 3 5 から変更 1 ～ 3 を受信した。インポータ 9 1 5 は、変更 1 ～ 3 を用いて入力テーブル 9 2 5 を更新した。第 2 の段階 9 0 2 は、物理制御器 9 4 0 が完全なトランザクションを形成する変更 4 ～ 6 を送出していることを更に示す。制御器 9 4 5 は、完全なトランザクションを形成する変更 7 及び 8 を送出している。変更 4 ～ 6 及び変更 7 ～ 8 は、物理制御器 9 4 0 及び 9 4 5 が論理制御器 9 1 0 から汎用要求を受信したのに応答して作成した汎用応答をそれぞれ含む。いくつかの実施形態において、変更 4 ～ 8 は汎用要求の識別子を更に含む。

10

【0 1 2 3】

第 3 の段階 9 0 3 において、論理制御器 9 1 0 は、物理制御器 9 4 0 からトランザクションの入力の変更 4 ～ 6 のセットを受信し、物理制御器 9 4 5 からトランザクションを形成する入力の変更 7 及び 8 を受信した。インポータ 9 1 5 は、変更 4 ～ 8 を用いて入力テーブル 9 2 5 を更新する。この時点で、入力テーブル 9 2 5 は変更 1 ～ 8 を有する。第 3 の段階 9 0 3 は、規則エンジン 9 2 0 がテーブルマッピング動作を実行して、前の段階 9 0 2 において入力テーブル 9 2 5 に入力された変更 1 ～ 3 から出力の変更を生成したことを更に示す。

20

【0 1 2 4】

また、段階 9 0 3 において、論理制御器 9 1 0 は、(1) 論理制御器 9 1 0 が完全なトランザクションを形成する入力の変更を処理することにより出力の変更を生成したため、変更 1 ～ 3 を処理することにより生成されたこれらの出力の変更を送出するか、あるいは(2) 汎用応答を含む更なる入力の変更が供給されるのを待つかを判定する。物理制御器は、特定の基準に基づいてこの判定を行う。例えばいくつかの実施形態において、論理制御器 9 1 0 は、論理制御器 9 1 0 から汎用要求を受信した物理制御器の全てが汎用応答を渡すのを待つ。これらの実施形態において、論理制御器 9 1 0 は、全ての汎用応答を処理することにより生成された出力の変更を集約して、論理応答を生成する。その代わりに又はそれに加えて、論理制御器 9 1 0 は、汎用要求が物理制御器に送出されてから所定の期間中に受信された汎用応答を処理することにより生成された出力の変更を集約する。論理制御器は、所定の期間中に集約された出力の変更から論理応答を生成する。

30

【0 1 2 5】

第 4 の段階 9 0 4 において、論理制御器 9 1 0 は、論理応答を含むトランザクションの出力の変更の単一セットを生成するために、汎用応答を含む更なるトランザクションを使用する必要があると判定した。従って、論理制御器 9 1 0 は、前の段階 9 0 3 において計算された入力の変更 1 ～ 3 から計算された出力の変更を送出していない。規則エンジン 9 2 0 は、変更 4 ～ 8 に対してテーブルマッピング動作を実行して、対応する出力の変更を生成した。しかし、示すように、この場合、入力の変更 4 ～ 8 から計算された出力の変更は、入力の変更 1 ～ 3 から計算された出力の変更と共にグループ化される。従って、論理制御器 9 1 0 は、3 つの完全なトランザクションを形成する入力の変更 1 ～ 3、4 ～ 6 及び 7 ～ 8 を集約することにより、論理応答を含むこの出力の変更の単一セットを生成した。

40

【0 1 2 6】

第 5 の段階 9 0 5 において、論理制御器 9 1 0 は、変更 1 ～ 8 から計算された出力の変更を入力変換制御器 (不図示) に送出した。物理制御器は、入力テーブル 9 2 5 から入力の変更 1 ～ 8 を除去し、出力テーブル 9 3 0 から出力の変更を除去した。

【0 1 2 7】

50

図9は、論理制御器による汎用応答の集約を示す。物理制御器によるカスタマイズ済応答の集約を示すために、図9に示す論理制御器及び物理制御器をそれぞれ物理制御器及びシャシ制御器に置換できることが当業者には認識されるだろう。

【0128】

図10は、制御器の階層における下位レイヤの下位制御器のセットからの応答セットを集約して階層における上位レイヤの上位制御器に渡す単一の応答を生成するためにいくつかの実施形態が実行する処理1000を概念的に示す。いくつかの実施形態において、処理1000は、図8及び図9の論理制御器810及び910と同様の中間制御器により実行される。すなわち、中間制御器は、(1)上位制御器から要求を受信し、(2)要求を下位制御器のセットに分散し、(3)下位制御器からの要求に対する応答を集約して単一の応答を生成し且つ上位制御器に渡す。いくつかの実施形態において、中間制御器が受信又は生成する要求及び応答は、完全なトランザクションを形成する変更(例えば、データ

10

【0129】

いくつかの実施形態において、受信側制御器は、入力変換制御器から論理要求を受信し、物理制御器のセットに汎用要求を送出し、物理制御器から汎用応答を受信し且つ入力変換制御器に論理応答を送出する論理制御器である。論理制御器により実行される際の処理1000を以下に説明する。しかし、いくつかの実施形態の受信側制御器は、論理制御器から汎用要求を受信し、被管理転送要素のセットにカスタマイズ済要求を送出するか又はシャシ制御器のセットに汎用要求を中継し、被管理転送要素からカスタマイズ済応答を受信するか又はシャシ制御器から汎用応答を受信し且つ論理制御器に汎用応答を送出する物理制御器であってもよい。

20

【0130】

処理1000は、入力変換制御器から論理要求を受信する(1005)ことから開始する。入力変換制御器は、いくつかの実施形態のネットワーク制御システムのユーザにより提供された入力データから論理要求を生成する。論理要求は、入力変換制御器を介してユーザが管理する論理転送要素の特定の情報を問い合わせる。処理1000において、論理要求を汎用要求に変換することにより汎用要求を計算する(1010)。

【0131】

次に、処理1000において、汎用要求の送出先である物理制御器のセットを識別する(1015)。物理制御器のセットを識別するために、処理1000において、最初に論理転送要素を実現する被管理転送要素のセットを識別し、その後被管理転送要素のセットのマスタ物理制御器を識別する。いくつかの実施形態において、これらのマスタ物理制御器は汎用要求を受信する必要がある。処理1000において、識別された各物理制御器に汎用要求を送出する(1015)。いくつかの実施形態では、処理1000において、論理要求の識別子を保持し、汎用要求に識別子を追加する。処理1000において、汎用応答と汎用要求及び論理要求とをマッチングするために、当該識別子が使用される。

30

【0132】

識別された物理制御器のセットに汎用要求を送出した後、処理1000において、物理制御器から汎用応答を受信する(1020)。また、1020において、処理1000は汎用応答を含む入力の変更を処理して(例えば、テーブルマッピング動作を実行して)出力の変更を生成する。

40

【0133】

次に、処理1000において、物理制御器から少なくとも1つの完全なトランザクションを受信したかを判定する(1025)。完全なトランザクションは、物理制御器からバリアを受信してから別のバリアを受信するまでの間に当該物理制御器から受信した入力の変更を含む。物理制御器からの完全なトランザクションは、汎用応答を含む。

【0134】

処理1000において、物理制御器から少なくとも1つの完全なトランザクション(例えば、少なくとも1つの完全な汎用応答)を受信していないと判定される場合(1025

50

）、処理１０００は１０２０に戻り、物理制御器から更なる入力の変更を受信する。

【０１３５】

処理１０００において、少なくとも１つの完全なトランザクションを受信したと判定される場合（１０２５）、処理１０００は１０３０に進み、特定の集約基準が満たされるかを判定する。異なる実施形態は、異なる集約基準を含む。例えばいくつかの実施形態において、特定の基準は、汎用要求を送出してから（１０１５）又は論理要求を受信してから（１００５）経過した期間を含む。この期間が経過した場合、特定の基準は満たされる。その代わりに又はそれに加えて、いくつかの実施形態において、特定の基準は、汎用要求を受信した物理制御器の全てから汎用応答が受信されるかを含む。これらの実施形態において、汎用要求を受信した物理制御器の全てから汎用応答が受信される場合、特定の基準は満たされる。

10

【０１３６】

処理１０００において、特定の基準が満たされないと判定される場合（１０３０）、１０２０に戻り、汎用応答の受信を続行し、汎用応答を処理する。処理１０００において、特定の基準が満たされると判定される場合（１０３０）、受信した汎用応答（すなわち、汎用応答を含む完全なトランザクションの入力の変更）から計算された出力の変更を集約する（１０３５）。また、１０３５において、処理１０００は集約された出力の変更から単一の論理応答を生成する。

【０１３７】

その後、処理１０００において、論理制御器に論理要求を送出した入力変換制御器に論理応答を送出する（１０４０）。いくつかの実施形態の処理１０００において、出力の変更の末尾にバリアを挿入するか、あるいは出力の変更のうちの最後の変更に完全なトランザクションを示す情報を追加する。また、送出された出力の変更を論理制御器の出力テーブルから除去し、論理応答の計算に使用された完全なトランザクションを形成する入力の変更を論理制御器の入力テーブルから除去する。その後、処理は終了する。

20

【０１３８】

III. 電子システム

上述の特徴及びアプリケーションの多くは、コンピュータ可読記憶媒体（コンピュータ可読媒体とも呼ぶ）に記録された命令セットとして指定されるソフトウェア処理として実現される。１つ以上の処理装置（例えば、１つ以上のプロセッサ、プロセッサのコア又は他の処理装置）により実行される場合、これらの命令は、命令において示される動作を処理装置に実行させる。コンピュータ可読媒体の例は、ＣＤ－ＲＯＭ、フラッシュドライブ、ＲＡＭチップ、ハードドライブ、ＥＰＲＯＭ等を含むがそれらに限定されない。コンピュータ可読媒体は、無線又は有線接続を介する搬送波及び電子信号を含まない。

30

【０１３９】

本明細書において、用語「ソフトウェア」は、プロセッサにより処理するためにメモリに読み込み可能である読出し専用メモリに常駐するファームウェア又は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図する。また、いくつかの実施形態において、複数のソフトウェア発明は、残りの部分が異なるソフトウェア発明である大きいプログラムの一部として実現可能である。いくつかの実施形態において、複数のソフトウェア発明は、別個のプログラムとしても実現可能である。最後に、本明細書中で説明するソフトウェア発明を共に実現する別個のプログラムの何らかの組み合わせは、本発明の範囲内である。いくつかの実施形態において、１つ以上の電子システムにおいて動作するためにインストールされる場合、ソフトウェアプログラムは、ソフトウェアプログラムの動作を実行する１つ以上の特定のマシンの実現を定義する。

40

【０１４０】

図１１は、本発明のいくつかの実施形態が実現される電子システム１１００を概念的に示す。電子システム１１００は、上述の制御、仮想化又はオペレーティングシステムのアプリケーションのいずれかを実行するために使用可能である。電子システム１１００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

50

トコンピュータ、サーバコンピュータ、大型コンピュータ、ブレードコンピュータ等)、電話、PDA又は他の何らかの種類の電子装置であってもよい。そのような電子システムは、種々のコンピュータ可読媒体と、種々の他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1100は、バス1105、処理装置1110、システムメモリ1125、読出し専用メモリ1130、固定記憶装置1135、入力装置1140及び出力装置1145を含む。

【0141】

バス1105は、電子システム1100の多くの内部装置に通信可能に接続する全てのシステムバス、周辺バス及びチップセットバスを一括して表す。例えばバス1105は、処理装置1110と、読出し専用メモリ1130、システムメモリ1125及び固定記憶装置1135とを通信可能に接続する。

10

【0142】

処理装置1110は、これらの種々の記憶装置から、本発明の処理を実行するために実行する命令及び処理するデータを検索する。異なる実施形態において、処理装置は単一のプロセッサであってもよく、あるいはマルチコアプロセッサであってもよい。

【0143】

読出し専用メモリ(ROM)1130は、処理装置1110及び電子システムの他のモジュールが必要とする静的データ及び命令を格納する。それに対して、固定記憶装置1135は、読出し書込み記憶装置である。当該装置は、電子システム1100の電源がオフである場合でも命令及びデータを格納する不揮発性記憶装置である。本発明のいくつかの実施形態は、固定記憶装置1135として大容量記憶装置(磁気ディスク又は光ディスク及びそれに対応するディスクドライブ等)を使用する。

20

【0144】

他の実施形態は、固定記憶装置として取外し可能記憶装置(フロッピディスク、フラッシュドライブ等)を使用する。固定記憶装置1135と同様に、システムメモリ1125は読出し書込み記憶装置である。しかし、記憶装置1135と異なり、システムメモリは、ランダムアクセスメモリ等の揮発性読出し書込みメモリである。システムメモリは、プロセッサが実行時に必要とする命令及びデータの一部を格納する。いくつかの実施形態において、本発明の処理は、システムメモリ1125、固定記憶装置1135及び/又は読出し専用メモリ1130に格納される。処理装置1110は、これらの種々の記憶装置から、いくつかの実施形態の処理を実行するために実行する命令及び処理するデータを検索する。

30

【0145】

バス1105は、入力装置1140及び出力装置1145に更に接続する。入力装置により、ユーザは電子システムに対して情報を通信し且つコマンドを選択できる。入力装置1140は、英数字キーボード及びポインティングデバイス(「カーソル制御装置」とも呼ぶ)を含む。出力装置1145は、電子システムにより生成される画像を表示する。出力装置は、プリンタ、並びにブラウン管(CRT)又は液晶ディスプレイ(LCD)等の表示装置を含む。いくつかの実施形態は、入力装置及び出力装置の双方として機能するタッチスクリーン等の装置を含む。

40

【0146】

最後に、図11に示すように、バス1105は、ネットワークアダプタ(不図示)を介して電子システム1100をネットワーク1165に更に結合する。このように、コンピュータは、コンピュータのネットワーク(ローカルエリアネットワーク(「LAN」)、ワイドエリアネットワーク(「WAN」)又はイントラネット等)の一部であってもよく、あるいはインターネット等のネットワークの一部であってもよい。電子システム1100のいずれか又は全ての構成要素が本発明に関連して使用されてもよい。

【0147】

いくつかの実施形態は、マシン可読媒体又はコンピュータ可読媒体(あるいは、コンピュータ可読記憶媒体、マシン可読媒体又はマシン可読記憶媒体と呼ぶ)にコンピュータ

50

プログラム命令を格納するマイクロプロセッサ、記憶装置及びメモリ等の電子構成要素を含む。そのようなコンピュータ可読媒体のいくつかの例は、RAM、ROM、読出し専用コンパクトディスク（CD-ROM）、記録可能コンパクトディスク（CD-R）、書換え可能コンパクトディスク（CD-RW）、読出し専用デジタル汎用ディスク（例えば、DVD-ROM、2層式DVD-ROM）、種々の記録可能／書換え可能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）、磁気ハードドライブ及び／又は固体ハードドライブ、読出し専用Blue-Ray(r)ディスク及び記録可能Blue-Ray(r)ディスク、超高密度光ディスク、他の何らかの光媒体又は磁気媒体、並びにフロッピディスクを含む。コンピュータ可読媒体は、少なくとも1つの処理装置により実行可能であり且つ種々の動作を実行するための命令セットを含むコンピュータプログラムを格納してもよい。コンピュータプログラム又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、並びにインタープリタを使用するコンピュータ、電子構成要素又は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。

10

【0148】

上記の説明はソフトウェアを実行するマイクロプロセッサ又はマルチコアプロセッサを主に参照するが、いくつかの実施形態は、特定用途向け集積回路（ASIC）又はフィールドプログラマブルゲートアレイ（FPGA）等の1つ以上の集積回路により実行される。いくつかの実施形態において、そのような集積回路は、回路自体に格納される命令を実行する。

20

【0149】

本明細書中で使用される場合、用語「コンピュータ」、「サーバ」、「プロセッサ」及び「メモリ」は全て、電子装置又は他の技術装置を参照する。これらの用語は、人間又は人間のグループを含まない。本明細書の便宜上、用語「表示する」は、電子装置上に表示することを意味する。本明細書中で使用される場合、用語「コンピュータ可読媒体」及び「マシン可読媒体」は全て、コンピュータが読み出せる形態で情報を格納する有形の物理物体に限定される。これらの用語は、無線信号、有線ダウンロード信号及び他の何らかの一時的な信号を含まない。

【0150】

多くの特定の詳細を参照して本発明を説明したが、本発明が本発明の主旨から逸脱せず、に他の特定の形態で実現可能であることが当業者には認識されるだろう。更に、多くの図面（図4、図7及び図10を含む）は処理を概念的に示す。これらの処理の特定の動作は、図示及び説明される正確な順序で実行されなくてもよい。特定の動作は、一連の連続する動作で実行される必要がなく、異なる実施形態において、異なる特定の動作が実行されてもよい。更に、処理は、複数の副処理を使用して実現されてもよく、あるいは大きいマクロ処理の一部として実現されてもよい。

30

【図 1】

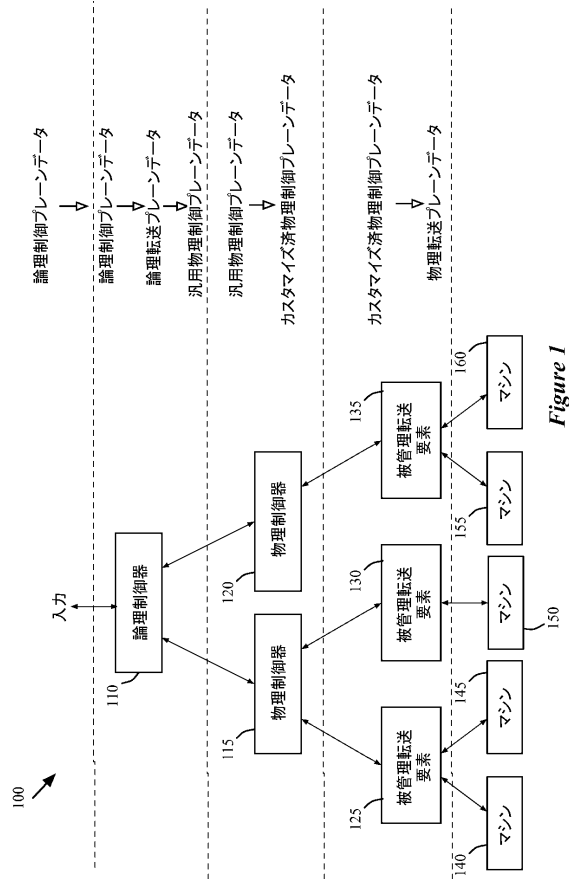


Figure 1

【図 2】

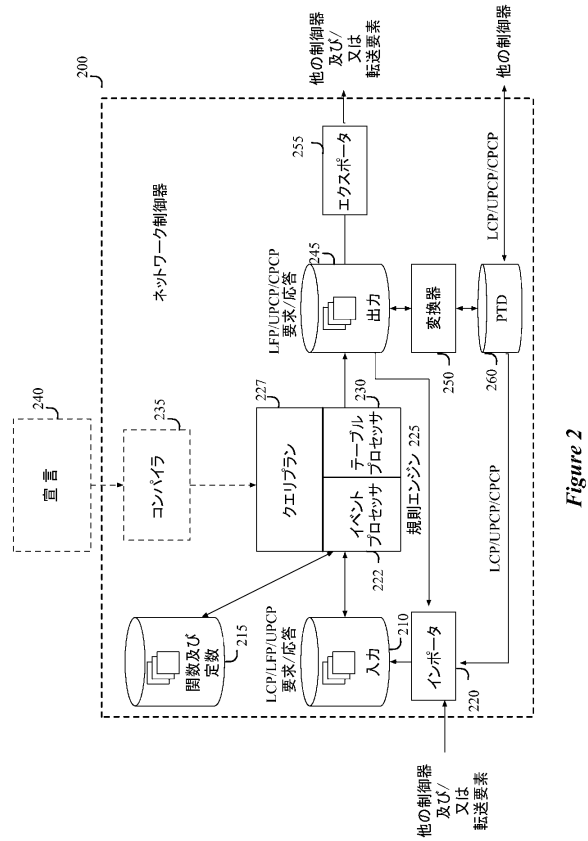


Figure 2

【図 4】

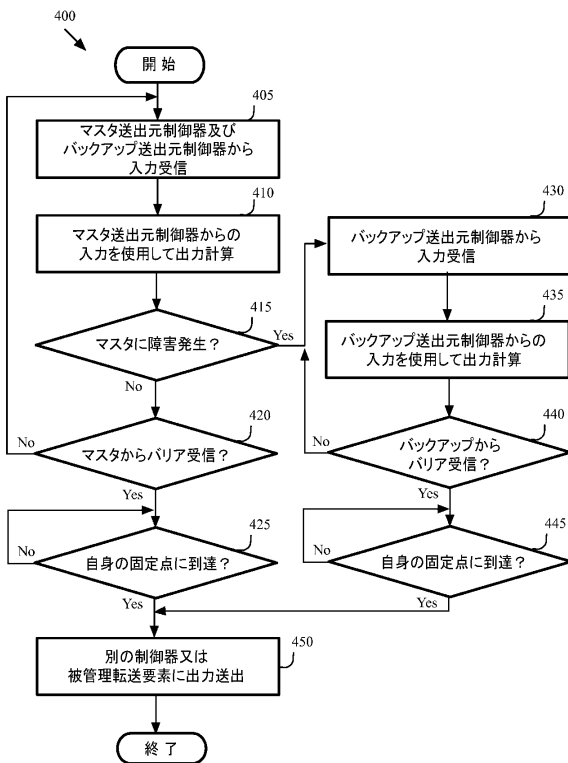


Figure 4

【図 5】

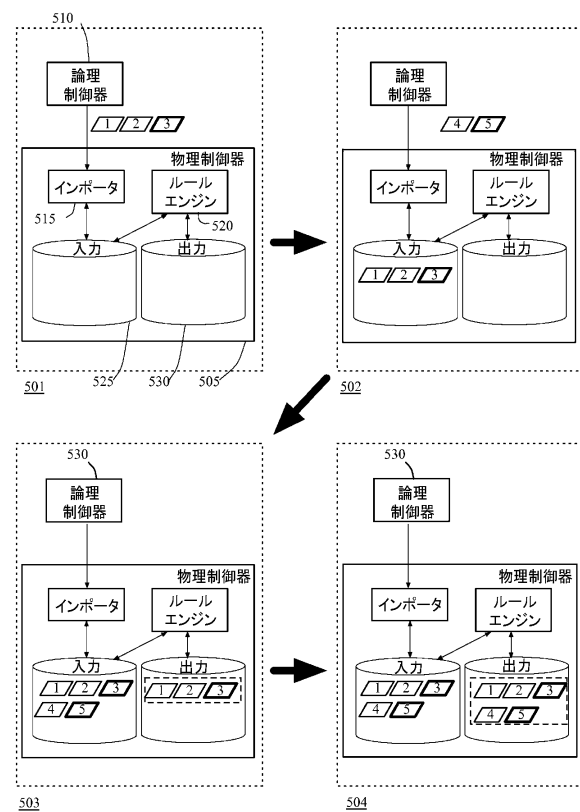


Figure 5

【図 6】

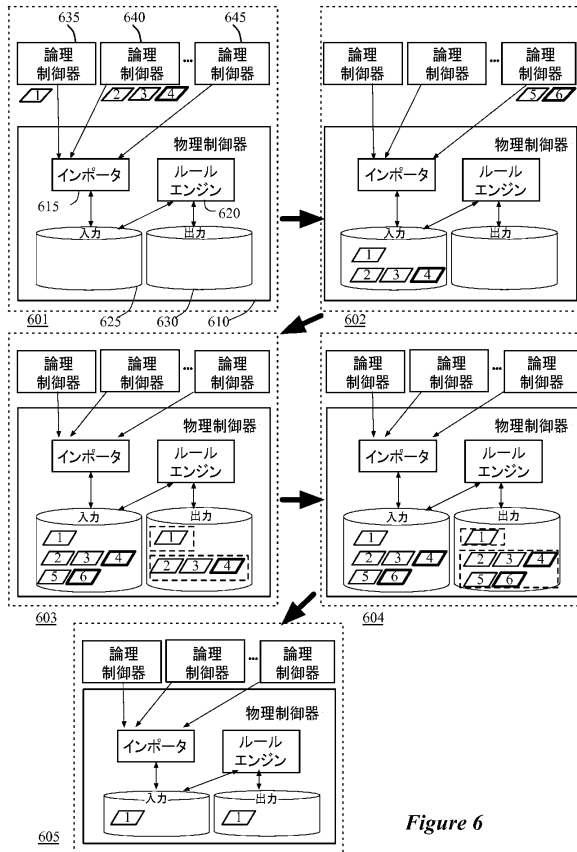


Figure 6

【図 7】

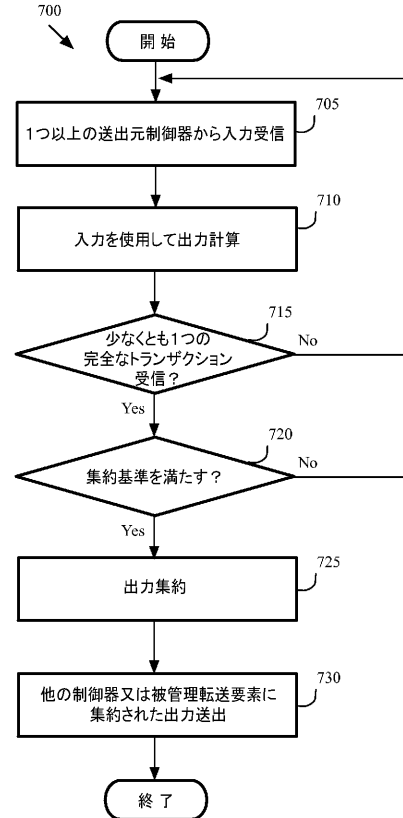


Figure 7

【图 8】

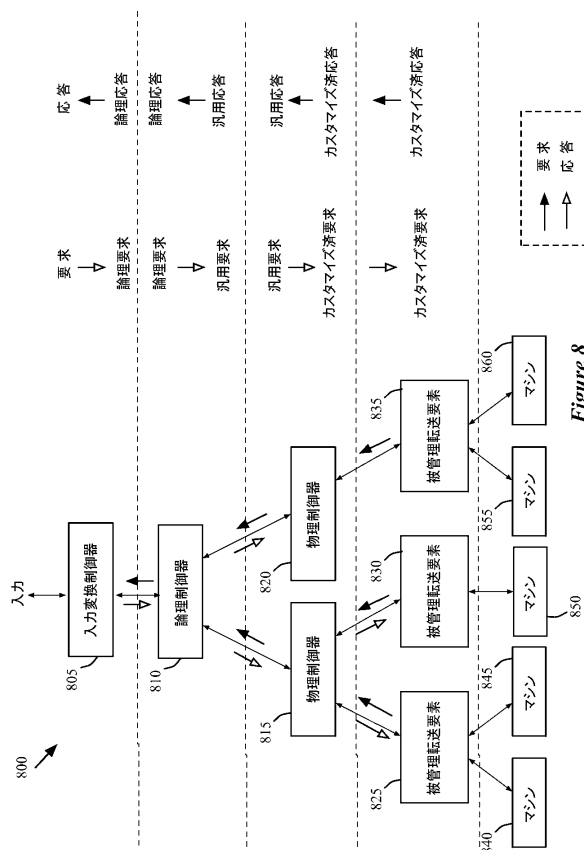


Figure 8

【图 9】

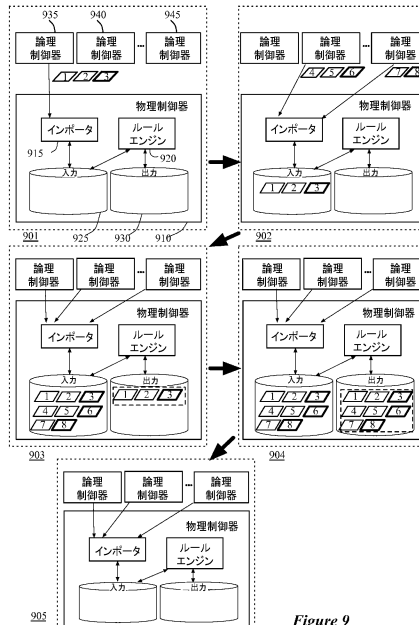


Figure 9

【図 10】

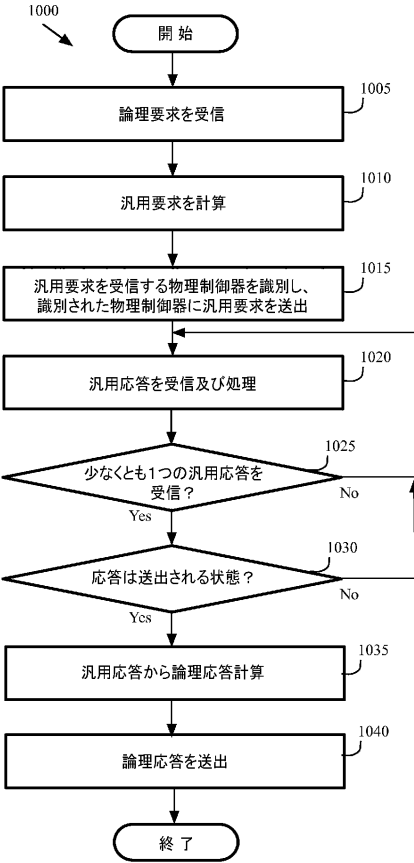


Figure 10

【図 11】

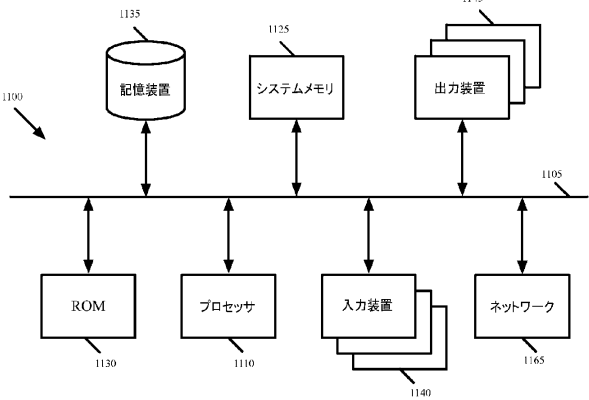


Figure 11

【図 3】

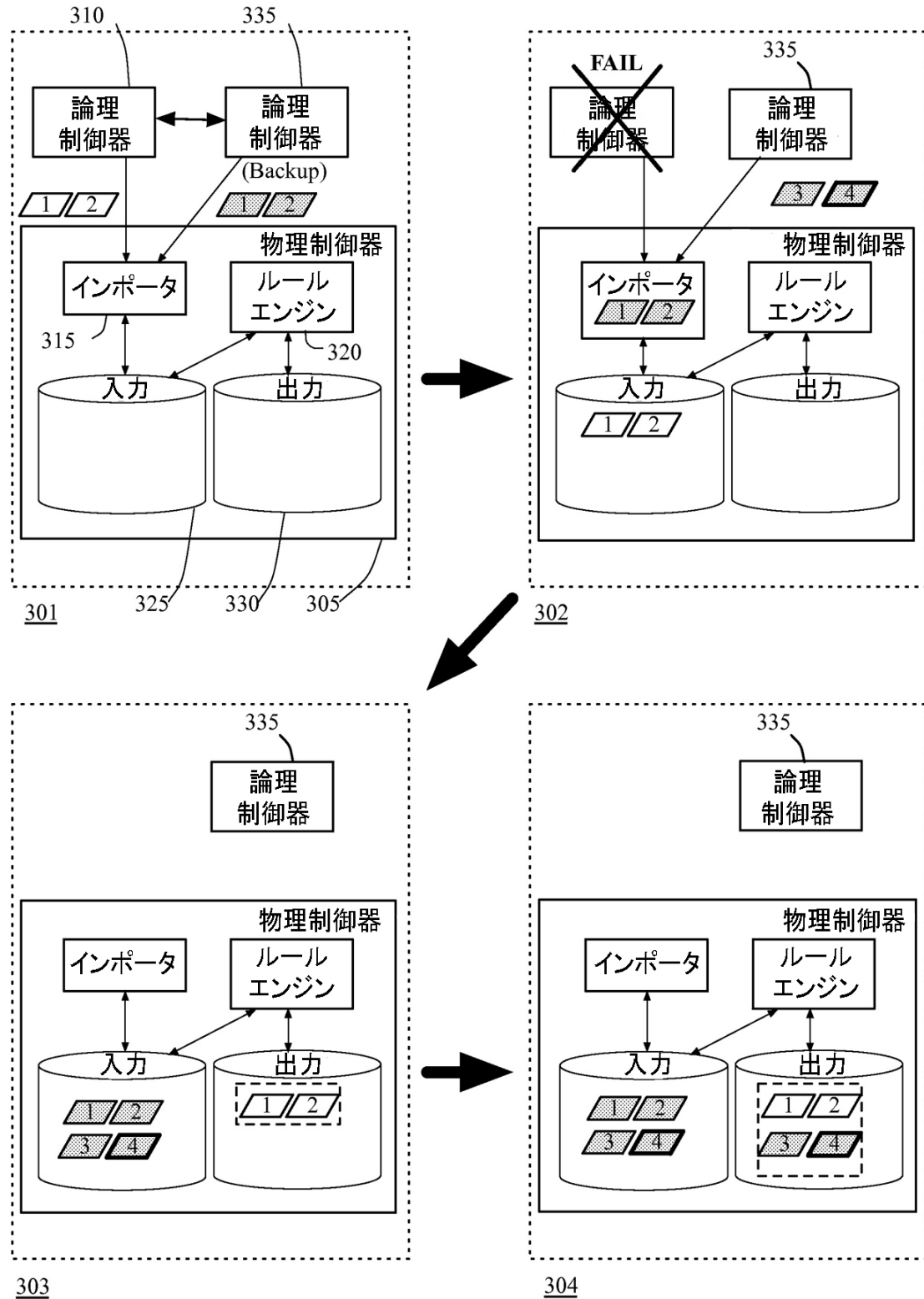


Figure 3

フロントページの続き

(31)優先権主張番号 61/635,056

(32)優先日 平成24年4月18日(2012.4.18)

(33)優先権主張国 米国(US)

(31)優先権主張番号 61/684,693

(32)優先日 平成24年8月17日(2012.8.17)

(33)優先権主張国 米国(US)

(72)発明者 コボネン, 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 ランベス, ダブリュー., アンドリュー

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 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

審査官 宮島 郁美

(56)参考文献 特開2009-159636 (JP, A)

特開平09-266493 (JP, A)

米国特許第05426774 (US, A)

米国特許出願公開第2004/0047286 (US, A1)

(58)調査した分野(Int.Cl., DB名)

H04L12/00-12/28, 12/44-12/955, 29/00-29/12

G06F13/00