

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第4541992号
(P4541992)

(45) 発行日 平成22年9月8日 (2010.9.8)

(24) 登録日 平成22年7月2日 (2010.7.2)

(51) Int.Cl. F I
G 0 6 F 13/00 (2006.01)
 G 0 6 F 13/00 3 5 3 V
 G 0 6 F 13/00 3 5 7 A

請求項の数 3 (全 20 頁)

(21) 出願番号	特願2005-224592 (P2005-224592)	(73) 特許権者	000001007
(22) 出願日	平成17年8月2日 (2005.8.2)		キヤノン株式会社
(65) 公開番号	特開2007-41803 (P2007-41803A)		東京都大田区下丸子3丁目30番2号
(43) 公開日	平成19年2月15日 (2007.2.15)	(74) 代理人	100076428
審査請求日	平成20年7月9日 (2008.7.9)		弁理士 大塚 康德
		(74) 代理人	100112508
			弁理士 高柳 司郎
		(74) 代理人	100115071
			弁理士 大塚 康弘
		(74) 代理人	100116894
			弁理士 木村 秀二
		(72) 発明者	平田 隆
			東京都大田区下丸子3丁目30番2号 キ
			ヤノン株式会社内

最終頁に続く

(54) 【発明の名称】 ネットワーク機器及びその制御方法、及びプログラム

(57) 【特許請求の範囲】

【請求項 1】

複数種の通信ミドルウェアを有する通信管理手段と、
 ネットワークに存在する機器が有する通信ミドルウェアのケーパビリティ情報を含む機器ケーパビリティ情報を取得する取得手段と、

前記取得手段により取得した前記機器ケーパビリティ情報から判断される、前記ネットワークに存在する機器と共通して使用できる通信プロトコルを基に、前記通信管理手段の複数種の通信ミドルウェアの内の活性化する通信ミドルウェアを判定する判定手段と、

前記判定手段により活性化すると判定された通信ミドルウェアを使用して前記ネットワークを介して機器間での通信を実行する通信手段と、

前記通信手段が、前記判定手段により活性化すると判定された通信ミドルウェアを使用した、前記ネットワークの他の機器との間の通信に失敗した場合、前記他の機器で用いられている通信プロトコルと、前記通信手段が前記他の機器との間で失敗した通信で使った通信ミドルウェアが用いた通信プロトコルとの差分情報を抽出する抽出手段と、

前記差分情報に応じて、前記他の機器と通信を実行するために前記通信手段が使用する通信ミドルウェアが用いる通信プロトコルを修正するプロトコル修正手段と、
 を有することを特徴とするネットワーク機器。

【請求項 2】

管理手段、取得手段、判定手段、通信手段、解析手段、及びプロトコル修正手段を具備するネットワーク機器の制御方法であって、

10

20

前記管理手段が、複数種の通信ミドルウェアを有する通信管理工程と、

前記取得手段が、ネットワークに存在する機器が有する通信ミドルウェアのケーパビリティ情報を含む機器ケーパビリティ情報を取得する取得工程と、

前記判定手段が、前記取得工程で取得した前記機器ケーパビリティ情報から判断される、前記ネットワークに存在する機器と共通して使用できる通信プロトコルを基に、前記複数種の通信ミドルウェアの内の活性化する通信ミドルウェアを判定する判定工程と、

前記通信手段が、前記判定工程により活性化すると判定された通信ミドルウェアを使用して前記ネットワークを介して機器間での通信を実行する通信工程と、

前記抽出手段が、前記ネットワークの他の機器との間で用いられている通信プロトコルと、自機で用いられる通信プロトコルとの差分情報を抽出する抽出工程と、

前記プロトコル修正手段が、前記差分情報に応じて、前記他の機器と通信を実行するために前記通信手段が使用する通信ミドルウェアが用いる通信プロトコルを修正するプロトコル修正工程と、

を有することを特徴とするネットワーク機器の制御方法。

【請求項3】

複数種の通信ミドルウェアを有する通信管理手順と、

ネットワークに存在する機器が有する通信ミドルウェアのケーパビリティ情報を含む機器ケーパビリティ情報を取得する取得手順と、

前記取得手順で取得した前記機器ケーパビリティ情報から判断される、前記ネットワークに存在する機器と共通して使用できる通信プロトコルを基に、前記複数種の通信ミドルウェアの内の活性化する通信ミドルウェアを判定する判定手順と、

前記判定手順により活性化すると判定された通信ミドルウェアを使用して前記ネットワークを介して機器間での通信を実行する通信手順と、

前記ネットワークの他の機器で用いられている通信プロトコルと、自機で用いられる通信プロトコルとの差分情報を抽出する抽出手順と、

前記差分情報に応じて、前記他の機器と通信を実行するために前記通信手段が使用する通信ミドルウェアが用いる通信プロトコルを修正するプロトコル修正手順と、
をコンピュータに実行させるためのプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、ネットワークに接続されるネットワーク機器及びその制御方法、及びプログラムに関する。

【背景技術】

【0002】

電子計算機技術の進歩により、従来は単一の機能しか有さず、相互に有機的に接続することが意図されていなかったオフィス機器や一般の家庭用機器においても、ネットワークを介して他の機器と接続され、共通する処理を実行するといった態様が提案されている。このような機器ネットワーク通じた融合を実現する技術として、UPnP (Universal Plug and Play), Jin i, J x t a等のプロトコルが知られている。

【0003】

UPnPは、HTTP技術やXML技術といった比較的高機能な処理を必要とする通信ソフトウェアで、これを各機器に対して常に個別に開発して搭載するのは容易ではない。そのため、こうしたUPnPなどの共通した通信部分のソフトウェアは、ミドルウェアという形で分離して実装するのが一般的となっている。また、通信機能を搭載した機器の高機能化により、こうしたミドルウェアを更に複数個同時に搭載したいという要求も出てきている。例えばUPnPにおいて、Jin iでも動的に発見できる通信機能をもった機器を実現するために、対応ミドルウェアを2種類搭載する必要がある。このように各種プロトコルに対応したい場合は、それぞれに対応した複数のミドルウェアを機器に搭載する必要がある。

【0004】

このようにして各機器が複数のプロトコルのそれぞれに対応したミドルウェアを搭載した場合、機器間で相互にデータをやり取りして処理を実行するには、同一のプロトコルが実装されているミドルウェアを搭載した機器を選択する必要がある。しかし、実装できるプロトコルの数には制限があり、世に存在するプロトコルを全て各機器に実装することは不可能である。また仮に実存するプロトコルを全て実装できたとしても、新たに開発されたプロトコルに対応することは不可能である。また各種プロトコルにおいて、その仕様には曖昧な部分があり全てが厳密に規定されているわけではない。このため、機器の開発者が、そのプロトコルの仕様に完全に従って実装したつもりでも、そのプロトコルの仕様の曖昧さの部分ため解釈の違いが生じ、機器間での通信が失敗する場合も起こりえる。

10

【0005】

また、プロトコルの仕様に完全に沿っていない状態でプロトコルを実装する場合も多々発生している。更に、仕様に沿っていないプロトコルを用いる機器が事実上の標準（デファクトスタンダード）となると、元々の仕様に沿ったプロトコルの機器を開発しても、事実上の標準となったプロトコルを用いる機器との通信が正常に行えない事態も発生する。通信が失敗する例として、例えば、ある実装では大文字／小文字を区別していないが、別の実装では大文字／小文字を区別するため通信が失敗する等のケースがある。また、SOAPなどのXML関連のプロトコルでは、ネームスペースの有無により通信が失敗する場合がある。

20

【特許文献1】特開平8-88672号公報

【非特許文献1】RFC2616, R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee "Hypertext Transfer Protocol--HTTP/1.1", <http://www.ietf.org/rfc/rfc2616.txt>, June, 1999

【発明の開示】

【発明が解決しようとする課題】

【0006】

高機能化、複雑化して肥大化していく通信用ミドルウェアを複数搭載することは、計算能力に制約のある機器の組み込み環境では大きな制約となっており、これを効率的に処理することが大きな課題となってきた。機器間で各機器のプロトコルによる通信を確保するために、プロトコルの違いを発見するプロトコルモニタに関する提案の1つに特許文献1がある。この特許文献1では、プロトコルの誤り箇所を検出し、その箇所を自動的に指摘することを可能としている。

30

【0007】

しかし、このような対応を出荷後の機器に対して行うのは容易ではない。また、そのようなプロトコルモニタを用いても、市場に存在する全ての機器に対処することは不可能であり、その検証が十分に行われることはない。そのためユーザは、機器の購入前に、その機器がどのような機器と通信を行えたかを、例えばインターネットでの報告を探すなどして選ばざるを得ない。

【0008】

本発明は、上記従来技術の問題点を解決することを目的とする。

40

【0009】

また本願発明の特徴は、通信機能を有する機器に搭載された通信ミドルウェアを、そのネットワークレベルでの最適化する技術を提供することにある。

【課題を解決するための手段】

【0010】

上記目的を達成するために本発明の一態様に係るネットワーク機器は以下のような構成を備える。即ち、

複数種の通信ミドルウェアを有する通信管理手段と、

ネットワークに存在する機器が有する通信ミドルウェアのケーパビリティ情報を含む機器ケーパビリティ情報を取得する取得手段と、

50

前記取得手段により取得した前記機器ケーパビリティ情報から判断される、前記ネットワークに存在する機器と共通して使用できる通信プロトコルを基に、前記通信管理手段の複数種の通信ミドルウェアの内の活性化する通信ミドルウェアを判定する判定手段と、

前記判定手段により活性化すると判定された通信ミドルウェアを使用して前記ネットワークを介して機器間での通信を実行する通信手段と、

前記通信手段が、前記判定手段により活性化すると判定された通信ミドルウェアを使用した、前記ネットワークの他の機器との間の通信に失敗した場合、前記他の機器で用いられている通信プロトコルと、前記通信手段が前記他の機器との間で失敗した通信で使
用した通信ミドルウェアが用いた通信プロトコルとの差分情報を抽出する抽出手段と、

前記差分情報に応じて、前記他の機器と通信を実行するために前記通信手段が使用する通信ミドルウェアが用いる通信プロトコルを修正するプロトコル修正手段と、を有することを特徴とする。

10

【0012】

上記目的を達成するために本発明の一態様に係るネットワーク機器の制御方法は以下の
ような工程を備える。即ち、

管理手段、取得手段、判定手段、通信手段、解析手段、及びプロトコル修正手段を具備
するネットワーク機器の制御方法であって、

前記管理手段が、複数種の通信ミドルウェアを有する通信管理工程と、

前記取得手段が、ネットワークに存在する機器が有する通信ミドルウェアのケーパ
ビリティ情報を含む機器ケーパビリティ情報を取得する取得工程と、

20

前記判定手段が、前記取得工程で取得した前記機器ケーパビリティ情報から判断される、
前記ネットワークに存在する機器と共通して使用できる通信プロトコルを基に、前記複
数種の通信ミドルウェアの内の活性化する通信ミドルウェアを判定する判定工程と、

前記通信手段が、前記判定工程により活性化すると判定された通信ミドルウェアを使用
して前記ネットワークを介して機器間での通信を実行する通信工程と、

前記抽出手段が、前記ネットワークの他の機器との間で用いられている通信プロトコ
ルと、自機で用いられる通信プロトコルとの差分情報を抽出する抽出工程と、

前記プロトコル修正手段が、前記差分情報に応じて、前記他の機器と通信を実行するた
めに前記通信手段が使用する通信ミドルウェアが用いる通信プロトコルを修正するプロ
トコル修正工程と、を有することを特徴とする。

30

【発明の効果】

【0014】

本発明によれば、通信機能を有する機器に搭載された通信ミドルウェアを、その機器が
接続されているネットワークレベルで最適化できる。

【発明を実施するための最良の形態】

【0015】

以下、添付図面を参照して本発明の好適な実施の形態を詳しく説明する。尚、以下の実
施の形態は特許請求の範囲に係る発明を限定するものでなく、また本実施の形態で説明さ
れている特徴の組み合わせの全てが発明の解決手段に必須のものとは限らない。

【0016】

40

図1は、本発明の実施の形態に係るネットワークシステム全体の構成図である。

【0017】

ここではネットワーク105に、デジタルカメラ100、デジタルビデオカメラ101、
プリンタ102、スキャナ103、プロジェクタ104が接続されている。尚、本実施
形態では上記機器を挙げているが、本発明はこれら機器に限定されるものではない。

【0018】

図2は、本実施の形態に係る機器における処理の概要を示した図である。本図において
は説明を容易に行う目的から、各処理部における具体的な機能を例として記してあるが、
本発明はこの構成に限定されるものではない。また、図2では本実施の形態に係る処理概
要を説明するために機器の構成を簡略化して示している。

50

【0019】

図2は、本発明の一実施の形態に係る機器の一例と、そのソフトウェアにおける内部構成を示した機能ブロック図である。

【0020】

カラープリンタ200は、本実施の形態に係る機能を搭載した機器であり、ここではネットワーク接続可能な高機能カラープリンタとする。このカラープリンタ200は、通常のプリンタと同様に、PCなどのクライアントからの文書データや印刷データを受信して、記録シートにカラー画像を印刷する。更に、印刷データの管理や再印刷、履歴管理などの付加機能を設けているものとする。このカラープリンタ200のソフトウェア構成としては、まず基本的な処理を提供するオペレーティングシステム及びファームウェア201が搭載されている。更にその上に、各共通機能を提供するミドルウェア202～205が搭載されている。更に最上部には、アプリケーション206が搭載されている。このアプリケーション206は、このカラープリンタ200に付加機能を提供するために用いられる。例えば前述の履歴管理などの処理を行うものである。本図では説明の便宜上アプリケーション206は1つだけを示しているが、複数存在してもかまわない。

10

【0021】

次にミドルウェア部分を詳細に説明する。

【0022】

202～205はネットワークにおける通信処理を実現するための通信ミドルウェアである。この通信ミドルウェアにおいて、UPnP対応ミドルウェア202、SLP対応ミドルウェア203、UDDI対応ミドルウェア204は、各々の通信プロトコルに対応した通信ミドルウェアである。更にこれらを統合してアプリケーション206から扱うためのインタフェースを提供するための通信管理ミドルウェア205を有している。

20

【0023】

図3は、このカラープリンタ200の電源がオンされて、その稼動が開始されたときに発生する処理を概念的に示した図である。

【0024】

まず、カラープリンタ200に内蔵されるオペレーティングシステム・ファームウェア201、アプリケーション206、各ソフトウェアにおけるケーパビリティ情報が収集される。そして、このカラープリンタ200に属したケーパビリティ情報である機器ケーパビリティ207が生成される。そして、その機器ケーパビリティ207を、ネットワーク105を通じて各機器から収集することにより、群ケーパビリティ208が構成される。この時、機器ケーパビリティ207から構成される群ケーパビリティ208は、どのような機能及び、どのような能力を有する機器が群を構成しているかによって生成される内容が変化する。例えば、デジタルカメラ100が存在する場合、このデジタルカメラ100が撮影した画像情報の印刷時における色再現性を高めるためのカラーマネージメント機能が必須と判断される。これにより群ケーパビリティ情報208としてカラーマネージメント機能を含むケーパビリティ情報が生成される。逆に、デジタルカメラが存在せずにモノクロ高速ドキュメントスキャナ103が存在する場合、高速なスキャン処理に追随するためのドラフトモードでのモノクロ印刷機能を含む群ケーパビリティ情報208が生成される。また、共通して使用できる通信プロトコルや対応できるデータフォーマットといった機能情報も群ケーパビリティ208として生成される。

30

40

【0025】

図4は、カラープリンタ200において群ケーパビリティ208と機器ケーパビリティ207とを用いた最適化処理によって、カラープリンタ200に内蔵されるいくつかの通信ミドルウェアが使用を停止した状態を示している。以下に、カラープリンタ200がこの状態に至るまでの処理の一例を以下に挙げる。

【0026】

まず、ネットワーク105に接続されているいずれの機器も通信手段（お互いの機器を検索する通信手段）としてUDDIを用いないことが、各機器の機器ケーパビリティから

50

算出された群ケーパビリティ 208 によって導き出されると仮定する。この場合は、各機器における UDDI 対応ミドルウェア 204 は使用停止となる。

【0027】

この時、アプリケーション 206 を含むカラープリンタ 200 が、この状況においてユーザに提供できる機能は変化していない点に注意が必要である。即ち、UDDI 対応ミドルウェア 204 は、現在のネットワーク上における機器構成では使用できないため、その稼動を停止してもカラープリンタ 200 を含む機器ネットワークがユーザに対して提供できる機能は一切変化しない。また、アプリケーション 206 は、引き続き通信処理において通信管理ミドルウェア 205 を通じて処理を行うので、UDDI 対応ミドルウェア 204 の稼動停止による影響を受けることはない。

10

【0028】

図 5 は、本発明の実施の形態に係る特徴を説明するブロック図である。

【0029】

図 5 において、プロトコル A に準拠している機器 A と、プロトコル A と B の両方に準拠している機器 B と、プロトコル B に準拠している機器 C が接続されている。ここで機器 A は機器 B の状態を監視するとともに、機器 B と機器 C の間の通信 505 で使用されるプロトコル B を監視する。更に、監視しているプロトコル B でのメッセージと、機器 B の状態を関連付けて記録する。更に、機器 B と機器 C の間のプロトコル B でのメッセージで遷移した機器 B の状態と、機器 A が使用可能なプロトコル A で同様の状態に遷移させることが可能なメッセージとを対応付ける。この対応付けを蓄積しておくことにより、機器 A はプロトコル B のメッセージの意味を解釈できるようになり、プロトコル B を模倣することが可能となる。よって、プロトコル B を用いて機器 A と機器 C との間での相互通信 507 が可能となる。

20

【0030】

図 6 は、本実施の形態に係るネットワークにおける通信処理を説明するブロック図である。

【0031】

このネットワークは、プロトコル A に準拠している複数の機器で構成されている。ここでは、プロトコル A に準拠している機器である機器 A と機器 B、機器 C とが接続されている。ここでは機器 B と機器 C との間で、プロトコル A の仕様の曖昧さにより生じた誤解によりプロトコル A には厳密に準拠していないプロトコル A' により相互運用性が確保されている。このとき機器 A は、機器 B から受信するメッセージ 604 と、機器 C から受信するメッセージ 606、更に機器 B と機器 C との間で通信されるメッセージ 605 を監視する。具体的には、機器 A のネットワークインタフェースをプロミスキャスモードで動作させて、ネットワーク 105 に流れる全パケットを受信して通信データを収集する。こうして収集した通信データを解析し、プロトコル A とプロトコル A' の差分を抽出し、機器 A が使用できるプロトコルに適用する。その結果、機器 A は、ローカルルールであるプロトコル A' で機器 B との通信 607 及び機器 C との通信 608 が可能となり、これら機器 A ~ C による相互運用性を確保することが可能となる。

30

【0032】

例えば、プロトコル A が「HTTP/1.1」（「非特許文献 1」）において、図 7 のメッセージをプロトコル A、図 9 のメッセージをプロトコル A' とする。このとき機器 A と機器 B の間の通信 604、及び機器 A と機器 C の間の通信 606 において、1 行目の文字列「GET」と「get」のような大文字と小文字の違いが発生している。この場合、図 8 に示すエラーメッセージを返す。

40

【0033】

これにより機器 A は、機器 B から受信するメッセージ 604 と、機器 C から受信するメッセージ 606、更に機器 B と機器 C との間で通信されるメッセージ 605 を監視して通信データを収集する。こうして収集した通信データを解析し、プロトコル A とプロトコル A' の差分（図 7 と図 9 の 1 行目の文字列「GET」と「get」の違い）を抽出し、機器 A が

50

使用できるプロトコルに適用する。その結果、機器Aは、ローカルルールであるプロトコルA' (図9) で、機器Bとの通信607及び機器Cとの通信608が可能 (リターン値として図10のメッセージが返却される) になる。こうして機器A~Cの3台での相互通信が可能となる。

【0034】

図11は、本実施の形態に係るネットワークシステム (プロファイル管理システム) に接続している機器の一例であるPC (パーソナルコンピュータ) の構成を説明するブロック図である。尚、これら機器は前述のカメラ100、101、プリンタ102、スキャナ103、プロジェクタ104や、このPC (パーソナルコンピュータ) 以外にも、ワークステーション、ノートブックPC、パームトップPC、コンピュータを内蔵したテレビ等の各種家電製品、通信機能を有するゲーム機、電話、FAX、携帯電話、PHS、電子手帳等を含む他の機器と通信するための通信機能を有する端末、又は、これらの組合せであっても良い。

【0035】

図11において、CPU1101は、このPC全体の制御を司る中央演算装置 (以下、CPU) である。RAM1102は、ランダムアクセスメモリで、CPU1101の主メモリとして、及び実行プログラムの領域や、そのプログラムの実行エリアならびにデータエリアを提供している。プログラムROM1103は、CPU1101の動作処理手順を記録しているリードオンリメモリである。このROM1103には、このPCの機器制御を行うシステムプログラムである基本ソフト (OS) を記録したプログラムROMとシステムを稼動するために必要な情報などが記録されたデータROMがある。尚、このROM1103の代わりに後述のHDD1109を用いる場合もある。ネットワークインターフェース (NETIF) 1104は、ネットワーク105を介して機器間でのデータ転送を行うための制御や接続状況の診断を行う。VRAM1105は、ビデオRAMで、このPCの稼動状態を示す表示部1106の画面に表示される画像データを展開し、その表示の制御を行う。表示部1106は、CRTや液晶などの表示ユニットを備える。キーボードコントローラ (KBC) 1107は、キーボードやポインティングデバイス等を含む入力装置1108からのキー信号等を入力してシステムバス1100に出力する。入力装置1108は、このPCの利用者が、このPCに対して行う操作を受け付けるための外部入力装置であり、例えば、キーボードなどである。HDD1109はハードディスク装置で、OSや各種アプリケーションプログラムがインストールされている。また画像情報などのデータ保存用にも用いられる。尚、本実施形態におけるアプリケーションプログラムとは、本実施形態を構成する各種手段を実行するソフトウェアプログラムなどである。FDD1110は、例えばフロッピー (登録商標) ディスクドライブ、CD-ROMドライブなどのリムーバブルな記憶媒体を装着して、データの読み出しや書込みを行うもので、上述したアプリケーションプログラムの媒体からの読み出しなどに用いられる。以下、FDDと記す。なお、HDD1109に格納するアプリケーションプログラムやデータをFDD1110に格納して使用することも可能である。システムバス1100は、上述した各ユニット間の接続するための入出力バス (アドレスバス、データバス、及び制御バス) である。

【0036】

図12は、本発明の実施の形態に係る機器に実装されるアプリケーションモジュール (機器ケーパビリティ管理システム) の機能ブロック図である。尚、このアプリケーションモジュールは、本実施の形態に係る機能を実現するソフトウェアのプログラムコードを記録した記録媒体を、システム或は装置に供給し、そのシステム或は装置のコンピュータ (CPUやMPU) が記憶媒体に格納されたプログラムコードを読み出して処理を実行することによって達成される。この場合、記録媒体から読み出されたプログラムコード自体が前述した実施形態の機能を実現することになり、そのプログラムコードを記録した記憶媒体が本発明の実施の形態に係る機能を構成することになる。

【0037】

また、コンピュータが読み出したプログラムコードを実行することにより、前述した実施形態の機能が実現されるだけでなく、そのプログラムコードの指示に基づき、コンピュータ上で稼動しているOS（オペレーションシステム）などが実際の処理の一部又は全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれる。

【0038】

更に、記憶媒体から読み出されたプログラムコードがコンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに書き込まれたあと、そのプログラムコードの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わるCPUなどが実際の処理の一部又は全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれる。

10

【0039】

この機器ケーパビリティ管理システムは、アプリケーション1200、機器ケーパビリティ管理部1201、メッセージ送受信部1210を有する。更に、機器ケーパビリティ管理部1201は、アプリケーション管理部1202、ソフトウェアケーパビリティ管理部1203、機器ケーパビリティ管理部1204、群ケーパビリティ管理部1205、ミドルウェア最適化部1206、ミドルウェア再構成部1207、ミドルウェア管理部1208、通信ミドルウェア1209、未知プロトコル学習部1220、既知プロトコル修正部1230を含む。また、メッセージ送受信部1210は、メッセージ送信部1211、メッセージ受信部1212を含む。

【0040】

20

図13は、本実施の形態に係る未知プロトコル学習部1220のモジュール構成を示すブロック図である。

【0041】

この未知プロトコル学習部1220は、学習コントロール部1221、仮想プロトコル部1222、テーブル読み出し書き込み部1223、プロトコル監視部1224、コマンドテーブル部1225、プロトコル記録テーブル部1226を含む。

【0042】

図14は、本実施の形態に係る既知プロトコル修正部1230のモジュール構成を示すブロック図である。

【0043】

30

この既知プロトコル修正部1230は、相互運用性判定部1231、プロトコルデータ収集部1232、差分抽出部1233、差分適用部1234を含む。

【0044】

尚、上述した図12～図14の各部における処理内容は以下の説明で適時参照して記述する。

【0045】

図15は、本実施の形態に係る未知プロトコル学習部1220のコマンドテーブル部1225の具体例を示す図である。

【0046】

このコマンドテーブル部1225は、IPアドレス（IPAddress）項目1501と、プロトコル適用可否（Existing）項目1502で構成されている。それぞれの項目に対して、検索、読み書き、追加、削除が可能であり、IPアドレス項目1501に対してプロトコル適用可否項目1502が一意に特定される。IPアドレス項目1501は、ネットワーク105に接続されている機器のIPアドレスを格納する。またプロトコル適用可否項目1502には、「True（適用可）」又は「false（適用不可）」の文字列が格納される。

40

【0047】

図16は、本実施の形態に係る未知プロトコル学習部1220のプロトコル記録テーブル部1226の具体例を示す図である。

【0048】

50

プロトコル記録テーブル部1226は、IPアドレス項目1601、メッセージ(Message)項目1602、前状態(beforeStat)項目1603、後状態(afterStat)項目1604、有効コマンド(existingCommand)項目1605を有している。それぞれの項目に対して、検索、読み書き、追加、削除が可能である。またIPアドレス項目1601に対して、メッセージ項目1602、前状態項目1603、後状態項目1604、有効コマンド項目1605が一意に特定される。IPアドレス項目1601は、ネットワーク105に接続されている機器のIPアドレスを格納する。メッセージ項目1602は、機器間でやり取りされ、ネットワーク105を流れるメッセージを格納する。前状態項目1603、後状態項目1604、はそれぞれネットワーク105に接続されている機器における、メッセージの受信前と受信後の状態を示す情報を格納する。有効コマンド項目1605は、前状態項目1603及び後状態項目1604に記載された状態を発生可能な、自機器に製品出荷時に予め実装されているプロトコルにおける対応コマンドが格納されている。

10

【0049】

図17は、本発明の一実施形態に係るネットワークシステム（機器ケーパビリティ管理システム）における処理を説明するフローチャートである。

【0050】

まずステップS1700で、このネットワークに接続されている各機器に搭載されている各々のソフトウェアに対するソフトウェアケーパビリティ情報を生成する。この処理はソフトウェアケーパビリティ管理部1203から、アプリケーション管理部1202及びミドルウェア管理部1208を通じて、アプリケーションソフトウェア、ミドルウェア、ファームウェアといった各機器に搭載されている各種のソフトウェアの情報をケーパビリティ情報1240にまとめることで行われる。次にステップS1701で、ステップS1700で収集したソフトウェアケーパビリティ情報1240から、各機器の単位でのケーパビリティ情報である機器ケーパビリティ207を生成する。そして、この機器ケーパビリティ207をネットワーク105を通じて他の機器に通知する。これは機器ケーパビリティ管理部1204において実施される。尚、この機器ケーパビリティ207を他の機器に通知する際には、インターネットプロトコル(IP)が用いられる。この場合IPのバージョンは、IPv4とIPv6の双方が使用可能である。また、実際に通信経路として使用される媒体は、有線或は無線を用いることができる。またプロトコルによっては、更にTCP, UDP, HTTP, SMTP, SNMP, FTPなどの通信プロトコルが用いられる場合もある。

20

30

【0051】

次にステップS1702で、ネットワーク105にある他の機器の機器ケーパビリティを取得し、これを処理することにより、このネットワークを構成する複数の機器からなる群に対して付与する群ケーパビリティ208を生成する。これは群ケーパビリティ管理部1205が実施する。次にステップS1703で、ステップS1702で求めた群ケーパビリティ208と、ステップS1701で求めた機器ケーパビリティ207とを比較・演算し、ミドルウェアの再構成情報を生成する。これはミドルウェア最適化部1206により実行される。

【0052】

40

次にステップS1704で、ミドルウェア最適化部1206によりステップS1703で生成されたミドルウェアの再構成情報を用いて、ミドルウェアの停止や開始を含む動的な再構成を実施する。これはミドルウェア再構成部1207で実施される。

【0053】

次にステップS1705で、ステップS1704で実施したミドルウェア再構成の結果に基づいて、起動すべき通信ミドルウェアが存在しているかどうかを判定する。このステップS1705で、起動すべき通信ミドルウェアが存在しないと判定した場合はステップS1710に進み、未知プロトコルの学習処理を実施する。これは、未知プロトコル学習部1220で実行される。尚、このステップS1710の処理は、図18を参照して詳しく後述する。

50

【0054】

一方、ステップS1705で起動すべき通信ミドルウェアが存在すると判定した場合はステップS1706に進み、その通信ミドルウェアが使用できるプロトコルが他の機器との通信に適用可能かどうかを判定する。この判定は、例えば、実際にそのプロトコルで通信を行い、その通信が成功するか失敗するかによって判定する。このステップS1706で適用できないと判定した場合はステップS1720に進み、既知プロトコル修正処理を実施する。これは、既知プロトコル修正部1230で実行される。尚、このステップS1720の処理は図20のフローチャートを参照して詳しく後述する。

【0055】

こうしてステップS1706、S1710、S1720のいずれかを実行するとステップS1707に進み、処理を継続するかどうかを判定する。ここで処理を継続すると判定した場合はステップS1700に進み、前述の処理を繰り返し実行する。またステップS1707で、処理を終了すると判定した場合は、この処理を終了する。

【0056】

図18は、図17のステップS1710の未知プロトコルの学習処理を示すフローチャートである。ここでは前述の図5を例にして、機器Aが機器BとCとの間でやり取りされる、機器Aにとって未知のプロトコル（プロトコルB）を学習する場合で説明する。

【0057】

まずステップS1711で、機器Aの未知プロトコル学習部1220の学習コントロール部1221が、監視対象である機器Bの状態をプロトコルAを用いて取得する。こうして取得した機器Bの状態を学習コントロール部1221が機器BのIPアドレスと共に、テーブル読み出し書き込み部1223に渡してプロトコル記録テーブル部1226に記録する。このテーブル読み出し書き込み部1223は、プロトコル記録テーブル部1226のIPアドレス項目1601と前状態項目1603に書き込みを行う。

【0058】

次にステップS1712で、学習コントロール部1221はプロトコル監視部1224に対して、機器Cから機器Bに送信されるプロトコルBのメッセージを監視させ、その監視結果の記録を指示する。ここで機器Cから機器Bに対して何らかのメッセージが送信されるとプロトコル監視部1224は、そのメッセージを獲得して保持し学習コントロール部1221に受け渡す。このメッセージを受け取った学習コントロール部1221は、テーブル読み出し書き込み部1223に対して、その書き込みを指示する。これによりテーブル読み出し書き込み部1223は、プロトコル記録テーブル部1226の機器BのIPアドレスに対応するメッセージ項目1602に、その取得したメッセージを書き込む。

【0059】

次にステップS1713で、学習コントロール部1221は、先と同じように、機器Bの状態をプロトコルAを用いて取得する。こうして取得した機器Bの状態を学習コントロール部1221が機器BのIPアドレスと共に、テーブル読み出し書き込み部1223に渡してプロトコル記録テーブル部1226に記録する。即ち、テーブル読み出し書き込み部1223は、プロトコル記録テーブル部1226のIPアドレス項目1601にある同じIPアドレスの後状態項目1604に、その取得した機器Bの状態を書き込む。

【0060】

図16の例で説明すると、1610のIPアドレス「192.168.0.11」が機器BのIPアドレスに対応しているとする。1611は、機器Cから機器Bに送信されたメッセージで、1613はそのメッセージを受信する前の機器Bの状態を示している。この前状態項目1603では、プリントステータスが「アイドル」となっている。1613は、このメッセージを受信した後の機器Bの状態を記録しており、ここではプリントステータスが「印刷実行中」となっている。

【0061】

次にステップS1714で、監視処理を終了するか否かを判定し、継続と判定した場合はステップS1711の処理に移行するが、終了と判定した場合はステップS1715の

10

20

30

40

50

処理に移行する。ステップS1715では、学習コントロール部1221はテーブル読み出し書き込み部1223を用いてプロトコル記録テーブル部1226に記録したデータの読み取りを行う。こうして読み取ったデータから機器Bが、機器Cからメッセージを受け取る前の状態（前状態項目1603）と受け取った後の状態（後状態項目1604）とを比較して、既存のプロトコルAで、プロトコルBのコマンドと同じ役割を果たすコマンドを選択する。こうして求めたコマンドを、有効コマンド項目1605に記録するようにテーブル読み込み書き込み部1223に指示する。

【0062】

この結果、前述の図16の例では、1613には、プロトコルBのコマンドと同じ役割を果たすプロトコルAのコマンドとして「Print」が格納されている。

10

【0063】

次にステップS1716で、学習コントロール部1221は、テーブル読み出し書き込み部1223を用いて、機器BのIPアドレスをコマンドテーブル部1225のIPアドレス項目1501に格納し、プロトコル適用可否項目1502に「True」を記録する（図15参照）。更に、学習コントロール部1221は、機器CのIPアドレスをコマンドテーブル部1225のIPアドレス項目1501に格納し、プロトコル適用可否項目1502に「false」を記録する。

【0064】

以上の処理により、機器Aは、アプリケーションからの要求を受付けると送信先のIPアドレスを取得し、テーブル読み出し書き込み部1223を用いて、コマンドテーブル部1225のIPアドレス項目1501にある同一の値があるレコードからプロトコル適用可否項目1502の値を取得する。ここでプロトコル適用可否項目1502の値が「True」の場合には既存プロトコルAを使用すると判断し、プロトコル適用可否項目1502の値が「false」の場合には、プロトコルBを使用すると判断して、学習コントロール部1221にコマンドを受け渡す。

20

【0065】

こうして、そのコマンドと、対応するIPアドレスを受け取った学習コントロール部1221は、テーブル読み出し書き込み部1223を用いてプロトコル記録テーブル1226を検索する。そして、IPアドレス項目1601の値と同じIPアドレスのレコードから、更に有効コマンド項目1605に格納されているコマンドと同じコマンドのレコードを導き出す。こうして導き出したレコードのメッセージ項目1602の値を取得し、仮想プロトコル部1222に受け渡す。これにより機器Aの仮想プロトコル部1222は、機器Cに対して、そのメッセージ項目1602のメッセージを送信する。尚、このメッセージを送信する際には、インターネットプロトコル（IP）が用いられる。この場合IPのバージョンはIPv4とIPv6の双方が使用可能である。また、実際に通信経路として使用される媒体は、有線あるいは無線を用いることができる。プロトコルによっては、更にTCP、UDP、HTTP、SMTP、SNMP、FTPなどの通信プロトコルが用いられる場合もある。こうして機器Aと機器CとがプロトコルBで通信を行うことができる。

30

【0066】

図19は、ディスプレイ（表示ユニット）（機器B）とデジタルカメラ（機器C）との間での画像データの転送シーケンス（プロトコルB）を説明する図である。この場合も、機器AはプロトコルAを有しており、機器Bの状態をモニタして、機器Cとの間でプロトコルBによる通信を行うことを意図しているものとする。

40

【0067】

まず1901の以前に、機器Aは機器Bが「アイドル」状態にあることを認識しているものとする。次に1901で、機器Cから機器Bに対してジョブがある旨が通知される。そして、その要求に対して機器Bが「OK」で応答する際（1902）、機器Bの状態は「アイドル」から「レディ」状態になる。これにより機器Aは、機器Bと機器Cとの間でのプロトコルBによるメッセージ「CreateJpb」を獲得する。また機器Aは、機器Bとの

50

間でのプロトコルAにより、機器Bの状態が「レディ」になったことを取得する。次に1903で、表示すべきデータ（メッセージ）が機器Cから機器Bに送信される。これにより機器Bの状態は、「レディ」からデータを受信したモード「Received（受信済）（画像の表示開始）」に移行する。このとき機器Aは、機器Bと機器Cとの間でのプロトコルBによるメッセージ「Send」を獲得する。更に機器Aは、機器Bとの間でプロトコルAにより機器Bの状態が「Received（受信済）（画像の表示開始）」になったと認識する。

【0068】

これら情報を取得してプロトコル記録テーブル部1226に記憶している機器Aでは、ディスプレイ（機器B）が「アイドル」状態から「レディ」状態に移行するプロトコルBのコマンド（1901）に対応する、プロトコルAのコマンドを求める。更に機器Aは、ディスプレイ（機器B）が「レディ」状態から「受信済」状態に移行するプロトコルBのコマンド（1903）に対応する、プロトコルAのコマンドを求める。そして、これらコマンドを使用して仮想プロトコルBを作成し、機器Aと機器C（デジタルカメラ）との通信を行うことができるようになる。即ち、機器Aは、機器Bが受信したメッセージの内容と、その機器Bの状態の変化に基づいて、プロトコルBで、データを送信するために用いられるコマンド（CreateJob、Send等）を知ることができる。

【0069】

従って、機器Aから機器C（デジタルカメラ）に画像を転送するには、まずプロトコルBの「CreateJob」に相当するメッセージを機器C（カメラ）に送信する。そのメッセージに対して「OK」で応答する際、機器C（デジタルカメラ）は「アイドル」状態から「レディ」状態に移行する。次に、保存すべきデータ（画像）が付与されたプロトコルBの「Send」に対応するメッセージが機器Aから機器Cに送信される。これにより機器Cの状態は、「レディ」からデータを受信したモード「Received（受信済）（画像の保存開始）」に移行する。即ち、機器Aは、プロトコルBを用いてデータを機器Cに画像を送信する場合、機器Bの状態の変化に基づいて学習したプロトコルBで用いられるデータを送信するためのコマンドを作成して機器Cに送信することができる。

【0070】

以上のように機器Aと機器Cの通信が、この仮想プロトコルBを用いることで可能となる。

【0071】

図20は、図17のステップS1720の既知プロトコルの修正処理を示すフローチャートである。

【0072】

まずステップS1721で、本実施の形態に係る既知プロトコルの修正部1230のプロトコルデータ収集部1232は、ネットワーク105を流れる全パケットを受信するためにメッセージ送受信部1210に対して、自身が管理するネットワークインタフェース1104をプロミスキャスモードで動作させるよう依頼する。次にステップS1722で、ネットワーク105に流れているメッセージをメッセージ受信部1212で受信する。尚、このメッセージ受信部1212でメッセージを受信する際には、インターネットプロトコル（IP）が用いられる。この場合IPのバージョンはIPv4とIPv6の双方が使用可能である。また、実際に通信経路として使用される媒体は、有線或は無線を用いることができる。プロトコルによっては、更にTCP、UDP、HTTP、SMTP、SNMP、FTPなどの通信プロトコルが用いられる場合もある。

【0073】

次にステップS1723で、プロトコルデータ収集部1232は、ステップS1722で受信したメッセージを解析する。次にステップS1724で、既知プロトコルの修正部1230の差分抽出部1233は、通信ミドルウェア1209が制御しているプロトコルと、ステップS1723でプロトコルデータ収集部1232が解析したデータとの間で差分情報を抽出する。このとき抽出される差分情報としては、例えば、プロトコルが「HTTP/1.1」（「非特許文献1」）の場合は、図7と図9の1行目の文字列「GET」と「get」の

ようなトリビアルな（些細な）差分を抽出する。

【0074】

こうして抽出された差分情報は、図21に示したテーブルの形態で記録される。

【0075】

図21は、差分抽出部1233により抽出した差分情報を記憶するテーブルの具体例を示す図である。

【0076】

ここでは監視時間、そのメッセージの送信元アドレス、送信先アドレス、送信元ポート、送信先ポート及び差分情報が記憶されている。この差分情報の例では、メッセージの1行目(1c1)に、プロトコルが「HTTP/1.1」で、文字列「GET」と「get」の違いがあることが記憶されている。

10

【0077】

次にステップS1725で、差分情報の抽出に成功したかどうかを判定し、抽出に成功したと判定した、この処理を終了する。一方、差分情報の抽出に失敗したと判定した場合はステップS1726に進み、データ収集及び差分抽出処理を継続するかどうかを判定する。このステップS1726での判定基準は、例えば、収集時間及び、収集回数などが所定時間或は所定回数になったかどうかによる。尚、この処理における判定基準はこれら2つの条件に限定されるものではなく、任意の判定基準を適用することが可能である。こうしてステップS1726で、データ収集及び差分抽出処理を継続すると判定した場合はステップS1722の処理に移行する。一方、ステップS1726で、データ収集及び差分抽出処理を終了すると判定した場合は、この処理を終了し、図17のステップS1707の処理に移行する。

20

【0078】

ここで例えば、制御コマンド「GET」を送信してエラーメッセージを受信した場合、送信した「GET」を所定の関係を有するメッセージを収集する（例えば、「GET」と大文字小文字の文字の種別の異なるコマンド「get」を収集）。こうしてこの差分情報（大文字小文字の文字種別の差）の抽出に成功すると（ステップS1722）、それ以降は、「GET」の代わりに「get」を送信する。

【0079】

また、コマンドに付加されたパラメータが原因でエラーになった場合（例えば、受信側が、「GET」コマンドに付加されるパラメータとして、HTTPのGETメソッドとして使用されるHOST（インターネットホストとポート番号）とAccept-Language（レスポンスとして望む言語）を必須としているのにも拘わらず、送信側が、HOSTのみ（Accept-Languageが付加されない）がパラメータとして付加された「GET」コマンドを送信し、エラーになった場合）には、以下のようになる。

30

【0080】

送信側は、送信エラーになった「GET」コマンドと所定の関係を有するコマンドを収集する。ここでは、送信エラーになった「GET」コマンドとは、付加されるパラメータの異なる「GET」コマンド（即ち、パラメータとして「HOST」と「Accept-Language」が付加された「GET」コマンド）が収集される。このパラメータの差分の抽出に成功すると（ステップS1725）、それ以降は「HOST」のみが付加された「GET」コマンドの代わりに、「HOST」と「Accept-Language」が付加された「GET」コマンドを使用する。尚、「Accept-Language」として何を要求するかは、送信側に予め設定された「Accept-Language」を要求するようにすればよい。また、収集した「GET」コマンドに付加された「Accept-Language」の内容を表示し、その中から、使用者が所望の「Accept-Language」を選択するようにしてもよい。

40

【0081】

尚、送信側は、受信側で要求されているパラメータ（例えば「Accept-Language」）を自装置が有するか判断し、自装置が有さないパラメータであると判断すると、受信側が使用しているプロトコルは許容できないとして、受信側へのメッセージの送信を中止する。

50

また、コマンドの大文字／小文字と、コマンドに付加されたパラメータの両方に差異がある場合も同様に対応することができる。

【0082】

以上説明したように本実施の形態によれば、単一の機器内部での最適化に留まらず、通信機能を有する機器に搭載された通信ミドルウェアの機器ネットワークレベルでの最適化を実現することが可能となる。即ち、複数の通信ミドルウェアが搭載されている機器が集まって群を構成するとき、各機器内部で動的にその群の特性を生かした最適な通信ミドルウェアの組み合わせを構築することができるようになる。

【0083】

この時、各機器におけるアプリケーションは、この通信ミドルウェアの再構成を意識する必要がなく、また提供される機能も低下しない。これにより、高機能な通信ミドルウェアを複数搭載しながら、実行時に不要なものは使用しないようにできるため、計算機資源の有効活用ができる機器を実現することを可能となる。

10

【0084】

また起動すべき通信ミドルウェアが存在しないとき、即ち、機器が該当のプロトコルを用いた通信ができない場合に、既知のプロトコルを用いて未知のプロトコルのメッセージと、それを受けた機器の状態を対にして記憶し、未知のプロトコルに対して対処することが可能となる。

【0085】

更に、起動すべき通信ミドルウェアが存在した場合でも、仕様には曖昧さがあり厳密でない。このため、機器のベンダーが完全に仕様に沿ってプロトコルを実装したつもりでも仕様の曖昧さのために解釈の違いが生じ、通信が失敗する場合が起こりえる。このような場合でも、自機器が使用しているプロトコルと、接続しているネットワークで有効な他のプロトコルとの差分を抽出し、自機器のプロトコルを修正することにより、他の機器との通信を確保できる。

20

【0086】

以上説明したように本実施の形態によれば、以下のような効果がある。

(1) 単一の機器内部での最適化に留まらず、複数の機器を接続している機器ネットワークレベルで、通信機能を有する機器に搭載された通信ミドルウェアの最適化を実現できる。即ち、複数の通信ミドルウェアが搭載されている機器が集まって群を構成するとき、各機器内部で動的にその群の特性を生かした最適な通信ミドルウェアの組み合わせを構築することができる。この時、各機器内におけるアプリケーションは、この通信ミドルウェアの再構成を意識する必要がなく、また提供される機能も低下しない。これにより、高機能な通信ミドルウェアを複数搭載しながら、実行時に不要なミドルウェアは使用しないようにして、計算機の資源の有効活用ができる。

30

(2) 起動すべき通信ミドルウェアが存在しないとき、即ち、機器が該当のプロトコルを用いた通信ができない場合に、既知のプロトコルを用いて未知のプロトコルのメッセージと、それを受けた機器の状態を対にして記憶する。そして、その機器の状態を引き起こす既存のプロトコルにおける、対応するコマンドを用いることにより未知のプロトコルに対しても機器が対応できる。

40

(3) 機器のベンダーが仕様に完全に沿ってプロトコルを実装しているにも拘わらず、仕様の曖昧な部分で解釈の違いが発生して通信ができない場合であっても、自機が使用しているプロトコルと、対象プロトコルとの差分を抽出して自機のプロトコルを修正することにより、その機器との通信を可能にできる。

【図面の簡単な説明】

【0087】

【図1】本発明の実施の形態に係るネットワークシステム全体の構成図である。

【図2】本発明の一実施の形態に係る機器の一例と、そのソフトウェアにおける内部構成を示した機能ブロック図である。

【図3】実施の形態に係るカラープリンタの電源がオンされて、その稼動が開始されたと

50

きに発生する処理を概念的に示した図である。

【図4】カラープリンタにおいて群ケーパビリティと機器ケーパビリティとを用いた最適化処理によって、カラープリンタに内蔵される通信ミドルウェアが使用を停止した状態を示す図である。

【図5】本発明の実施の形態に係る特徴を説明するブロック図である。

【図6】本実施の形態に係るネットワークにおける通信処理を説明するブロック図である。

【図7】本実施の形態に係るプロトコルAのメッセージの一例を示す図である。

【図8】本実施の形態に係るエラーメッセージの一例を示す図である。

【図9】本実施の形態に係るプロトコルA'のメッセージの一例を示す図である。

【図10】本実施の形態に係るリターン値として返送されるメッセージ例を示す図である。

【図11】本実施の形態に係るネットワークシステム（プロファイル管理システム）に接続している機器の一例であるPC（パーソナルコンピュータ）の構成を説明するブロック図である。

【図12】本発明の実施の形態に係る機器に実装されるアプリケーションモジュール（機器ケーパビリティ管理システム）の機能ブロック図である。

【図13】本実施の形態に係る未知プロトコル学習部のモジュール構成を示すブロック図である。

【図14】本実施の形態に係る既知プロトコル修正部のモジュール構成を示すブロック図である。

【図15】本実施の形態に係る未知プロトコル学習部のコマンドテーブル部の具体例を示す図である。

【図16】本実施の形態に係る未知プロトコル学習部のプロトコル記録テーブル部の具体例を示す図である。

【図17】本発明の一実施形態に係るネットワークシステム（機器ケーパビリティ管理システム）における処理を説明するフローチャートである。

【図18】図17のステップS1710の未知プロトコルの学習処理を示すフローチャートである。

【図19】ディスプレイ（表示ユニット）（機器B）とデジタルカメラ（機器C）との間での画像データの転送シーケンス（プロトコルB）を説明する図である。

【図20】図17のステップS1720の既知プロトコルの修正処理を示すフローチャートである。

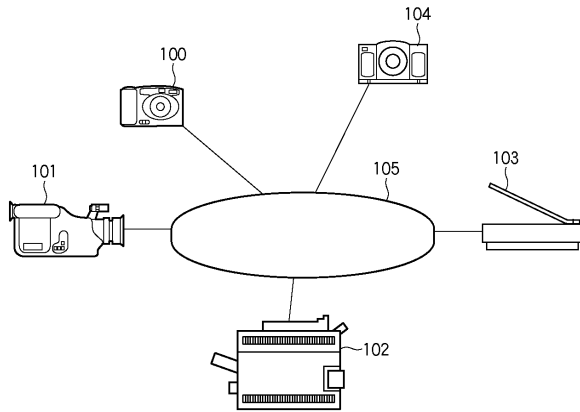
【図21】本実施の形態に係る差分抽出部により抽出した差分情報を記憶するテーブルの具体例を示す図である。

10

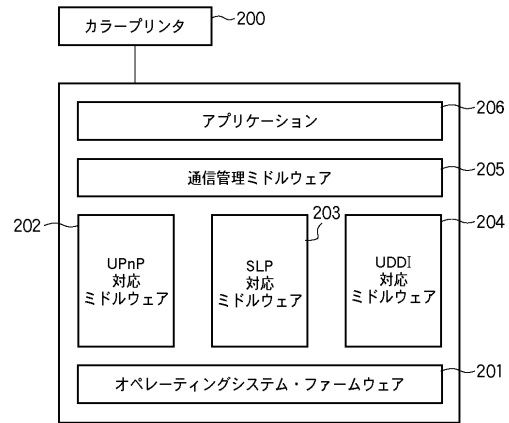
20

30

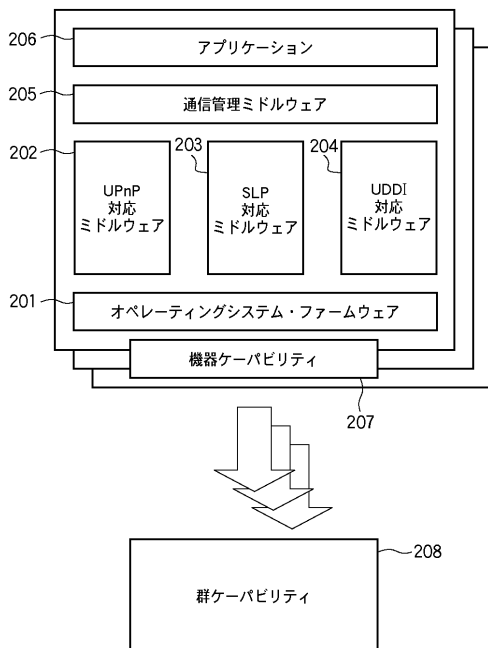
【図 1】



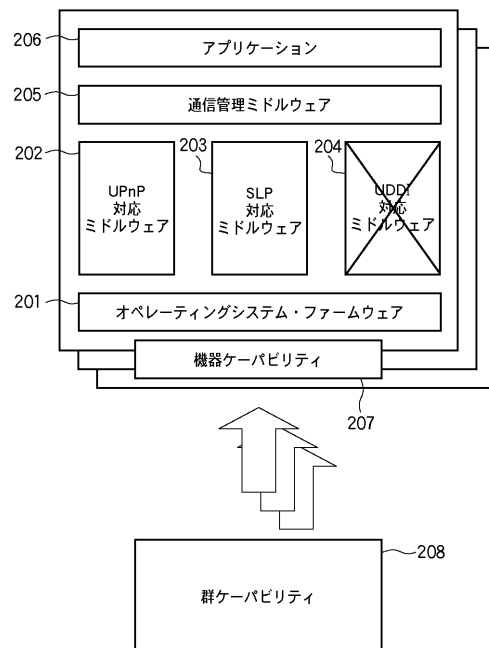
【図 2】



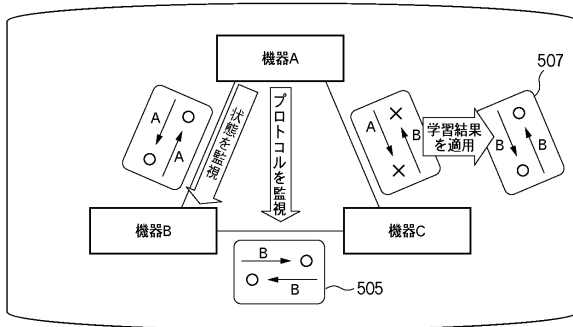
【図 3】



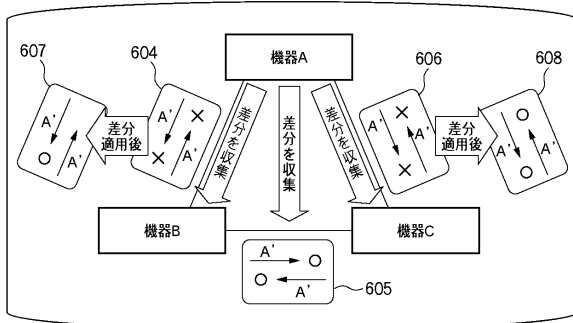
【図 4】



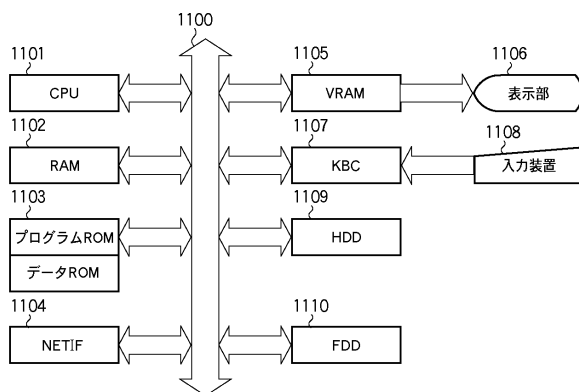
【図 5】



【図 6】



【図 11】



【図 7】

GET / HTTP / 1.1
 Accept-Language: ja
 User-Agent: OriginalBrowser / 1.0
 Connection : Keep-Alive

【図 8】

HTTP /1.1 400 Bad Request
 Date : Wed, 15 Dec 2004 10 : 00 : 01 GMT
 Server : OriginalServer / 1.0
 Content-Length : 0
 Connection : close

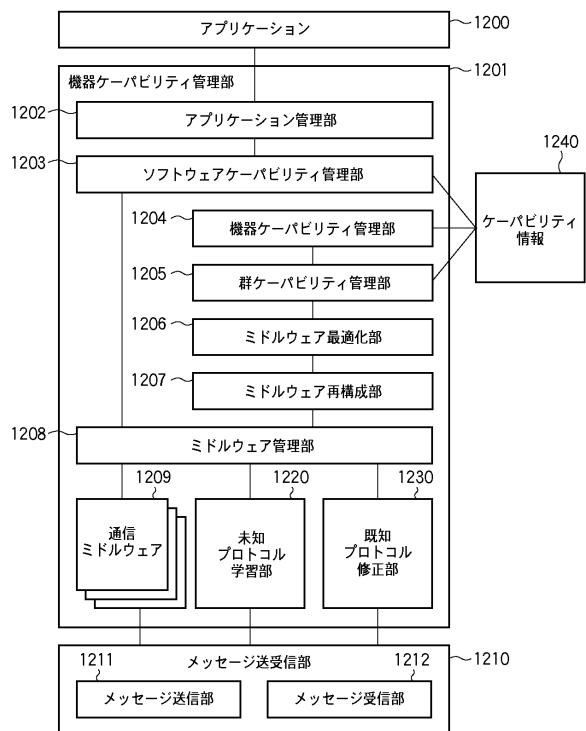
【図 9】

get / HTTP / 1.1
 Accept-Language : ja
 User-Agent : OriginalBrowser / 1.0
 Connection : Keep-Alive

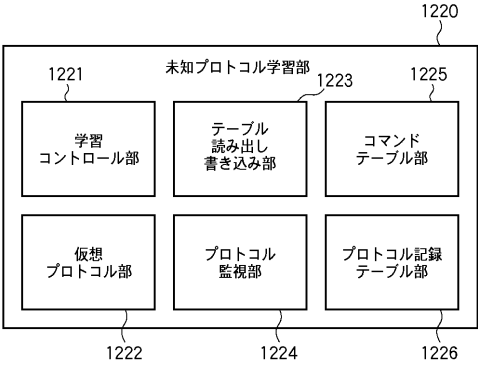
【図 10】

HTTP /1.1 200 OK
 Date : Wed, 15 Dec 2004 10 : 00 : 40 GMT
 Server : OriginalServer / 1.0
 Content-Length : 0
 Connection : close

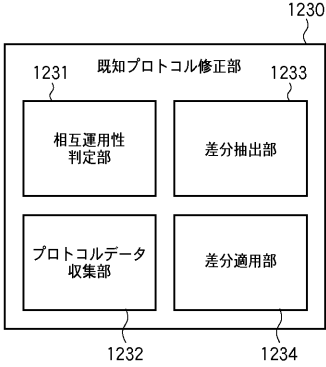
【図 12】



【図 1 3】



【図 1 4】



【図 1 5】

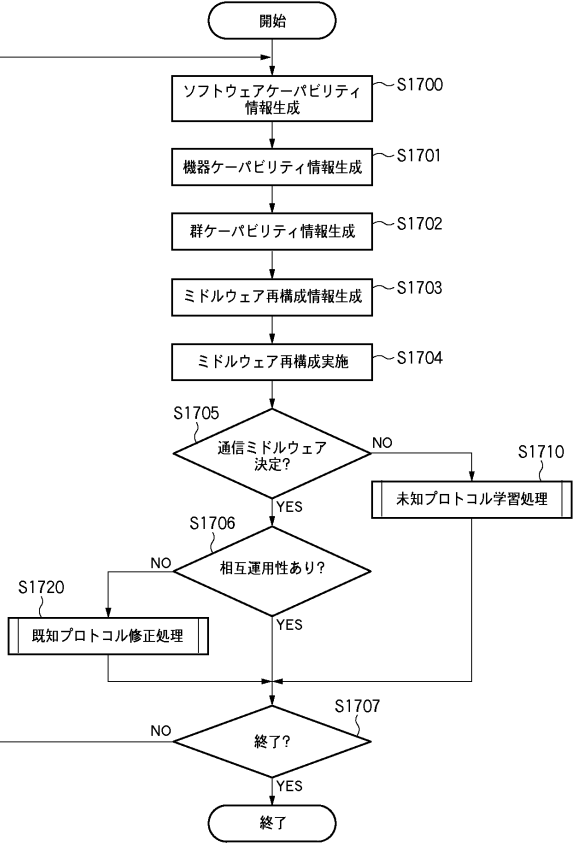
1501		1502
IP Address		Existing
192.168.0.11		True
192.168.0.12		True
192.168.0.22		false
192.168.0.19		false
192.168.0.114		True
192.168.0.14		false
192.168.0.36		false
192.168.0.20		True

1225

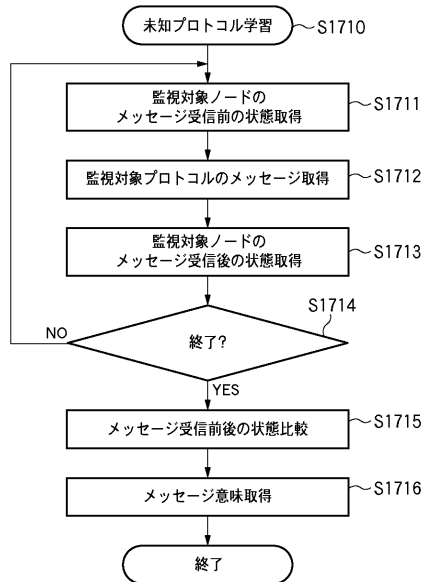
【図 1 6】

1605		1604		1603		1602		1601	
IP Address		Message		beforeStat		afterStat		existingCommand	
192.168.0.11		POST /control HTTP/1.1 HOST: 192.168.0.11 SOAPACTION: urn:schemas:upnp-org:service:PrintBasic:2#createdJob <soap:Envelope xmlns:soap=http://schemas.xmlsoap.org/soap/envelope/ xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance> <soap:Header> <Print> </Print> </soap:Header> <Print> </Print> </soap:Body> </soap:Envelope> </Print> </soap:Body> </soap:Envelope>		PrinterState=idle PrinterState=idle		PrinterState=processing PrinterState=processing		Print	
192.168.0.20		POST /control HTTP/1.1 HOST: 192.168.0.20 SOAPACTION: urn:schemas:upnp-org:service:Scan:1#scan <soap:Envelope xmlns:soap=http://schemas.xmlsoap.org/soap/envelope/ xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance> <soap:Header> <Scan> </Scan> </soap:Header> <Scan> </Scan> </soap:Body> </soap:Envelope>		idle		processing		scan	
192.168.0.12		POST /control HTTP/1.1 HOST: 192.168.0.12 SOAPACTION: urn:schemas:upnp-org:service:LightPowerOn:1#powerOn <soap:Envelope xmlns:soap=http://schemas.xmlsoap.org/soap/envelope/ xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance> <soap:Header> <PowerOn> </PowerOn> </soap:Header> <PowerOn> </PowerOn> </soap:Body> </soap:Envelope>		off		on		lightOn	

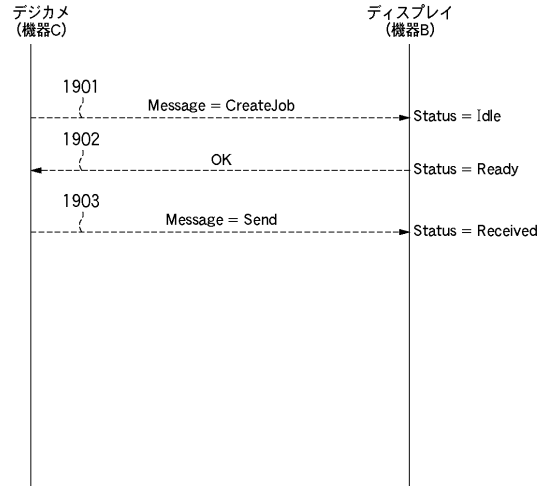
【図 1 7】



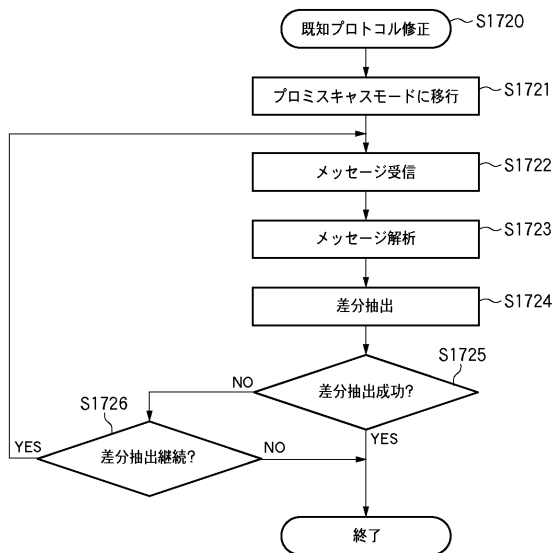
【図 18】



【図 19】



【図 20】



【図 21】

監視時間	Web, 15 Dec 2004 10:00:20
送信元アドレス	192. 168. 0. 3
送信先アドレス	192. 168. 0. 4
送信元ポート	12345
送信先ポート	80
差分	diff GetMessage 1cl < GET / HTTP / 1.1 --- >get / HTTP1.1

フロントページの続き

- (72)発明者 藤井 憲一
東京都大田区下丸子3丁目30番2号 キヤノン株式会社内
(72)発明者 下野 雅樹
東京都大田区下丸子3丁目30番2号 キヤノン株式会社内

審査官 木村 雅也

- (56)参考文献 特開平10-336191 (JP, A)
特開2002-152830 (JP, A)
特開2002-300226 (JP, A)

- (58)調査した分野(Int.Cl., DB名)
G06F 13/00
H04Q 7/38
H04L 12/28
H04L 29/06