

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号  
特許第4667175号  
(P4667175)

(45) 発行日 平成23年4月6日 (2011.4.6)

(24) 登録日 平成23年1月21日 (2011.1.21)

(51) Int.Cl.

F I

G O 6 F 13/10 (2006.01)

G O 6 F 13/00 (2006.01)

G O 6 F 3/12 (2006.01)

G O 6 F 13/10 3 3 0 B

G O 6 F 13/00 3 5 7 A

G O 6 F 3/12 C

請求項の数 9 (全 25 頁)

(21) 出願番号	特願2005-252472 (P2005-252472)	(73) 特許権者	000001007
(22) 出願日	平成17年8月31日 (2005.8.31)		キヤノン株式会社
(65) 公開番号	特開2007-66091 (P2007-66091A)		東京都大田区下丸子3丁目30番2号
(43) 公開日	平成19年3月15日 (2007.3.15)	(74) 代理人	100076428
審査請求日	平成20年8月26日 (2008.8.26)		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	西尾 雅裕
			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】 情報処理装置、情報処理装置の制御方法、並びに、コンピュータプログラム及びコンピュータ可読記憶媒体

(57) 【特許請求の範囲】

【請求項1】

ネットワーク通信手段を介してネットワークデバイスを検出し、当該検出したネットワークデバイスに対応するデバイスドライバを自動インストールする情報処理装置であって、

接続するネットワークを特定するネットワーク環境情報と当該ネットワーク環境情報で示されるネットワークに対して前記デバイスドライバの自動インストールのための機能の利用の可否を設定する設定手段と、

前記設定手段により設定された情報を記述した設定情報を記憶する記憶手段と、  
接続中のネットワークに関するネットワーク環境情報を取得する取得手段と、  
前記記憶手段により記憶されている設定情報の中から、前記取得手段で取得されたネットワーク環境情報に対応する設定情報を特定する特定手段と、

前記特定手段により特定された設定情報の内容を表示する表示手段と、  
前記特定手段により特定された設定情報に従い、接続中のネットワークのネットワークデバイスに対応するデバイスドライバの自動インストールの制御を行う制御手段とを備え、

前記デバイスドライバの自動インストールのための機能には、接続中のネットワーク上のネットワークデバイスから発行されたネットワーク参加メッセージを前記情報処理装置が搭載しているインストールの制御のためのシステムへ通知する第1機能と、ネットワークデバイスの探索メッセージをネットワークへ送信する第2機能とが含まれ、

10

20

前記設定手段は、ネットワーク環境情報に応じて前記第 1 機能及び前記第 2 機能のそれぞれの利用の可否を設定し、

前記制御手段は、前記設定情報において、前記第 1 機能の利用が可と設定されている場合には前記ネットワーク参加メッセージにより特定されるネットワークデバイスに対応するデバイスドライバの自動インストールを行い、前記第 1 機能の利用が否と設定されている場合には前記ネットワーク参加メッセージにより特定されるネットワークデバイスに対応するデバイスドライバの自動インストールを行わず、前記第 2 機能の利用が可と設定されている場合には前記探索メッセージにより発見されたネットワークデバイスに対応するデバイスドライバの自動インストールを行い、前記第 2 機能の利用が否と設定されている場合には前記探索メッセージにより発見されたネットワークデバイスに対応するデバイスドライバの自動インストールを行わないことを特徴とする情報処理装置。

10

【請求項 2】

前記ネットワーク環境情報は、DHCPを利用する／しない、自身のIPアドレス、サブネットマスク、デフォルトゲートウェイアドレス、DNSサーバアドレス、及びDHCPサーバアドレスの少なくともいずれかを含むことを特徴とする請求項 1 に記載の情報処理装置。

【請求項 3】

前記ネットワーク環境情報は、前記ネットワークに接続するために用いた通信機能の種類を含むことを特徴とする請求項 1 または 2 に記載の情報処理装置。

20

【請求項 4】

前記特定手段が前記取得手段で取得されたネットワーク環境情報に対応する設定情報が前記記憶手段になく、特定できなかった場合、前記設定手段により新規の設定情報に関する設定が行われることを特徴とする請求項 1 乃至 3 のいずれか 1 項に記載の情報処理装置。

【請求項 5】

更に、前記表示手段により設定情報の内容を表示した際に、ユーザの指示に応じて、当該設定情報の内容を編集する編集手段を備えることを特徴とする請求項 1 乃至 4 のいずれか 1 項に記載の情報処理装置。

【請求項 6】

ネットワークデバイスのデバイスドライバの自動インストールは、ネットワークプラグアンドプレイを利用し、前記ネットワーク参加メッセージはHelloメッセージ、前記探索メッセージはProbeメッセージであることを特徴とする請求項 1 に記載の情報処理装置。

30

【請求項 7】

ネットワーク通信手段を介してネットワークデバイスを検出し、当該検出したネットワークデバイスに対応するデバイスドライバを自動インストールする情報処理装置の制御方法であって、

接続するネットワークを特定するネットワーク環境情報と当該ネットワーク環境情報で示されるネットワークに対して前記デバイスドライバの自動インストールのための機能の利用の可否を設定する設定工程と、

40

前記設定手段により設定された情報を記述した設定情報を記憶手段に格納する格納工程と、

接続中のネットワークに関するネットワーク環境情報を取得する取得工程と、

前記記憶手段により記憶されている設定情報の中から、前記取得工程で取得されたネットワーク環境情報に対応する設定情報を特定する特定工程と、

前記特定工程により特定された設定情報の内容を表示する表示工程と、

前記特定工程により特定された設定情報に従い、接続中のネットワークのネットワークデバイスに対応するデバイスドライバの自動インストールの制御を行う制御工程とを有し、

前記デバイスドライバの自動インストールのための機能には、接続中のネットワーク上

50

のネットワークデバイスから発行されたネットワーク参加メッセージを前記情報処理装置が搭載しているインストールの制御のためのシステムへ通知する第 1 機能と、ネットワークデバイスの探索メッセージをネットワークへ送信する第 2 機能とが含まれ、

前記設定工程では、ネットワーク環境情報に応じて前記第 1 機能及び前記第 2 機能のそれぞれの利用の可否を設定し、

前記制御工程では、前記設定情報において、前記第 1 機能の利用が可と設定されている場合には前記ネットワーク参加メッセージにより特定されるネットワークデバイスに対応するデバイスドライバの自動インストールを行い、前記第 1 機能の利用が否と設定されている場合には前記ネットワーク参加メッセージにより特定されるネットワークデバイスに対応するデバイスドライバの自動インストールを行わず、前記第 2 機能の利用が可と設定されている場合には前記探索メッセージにより発見されたネットワークデバイスに対応するデバイスドライバの自動インストールを行い、前記第 2 機能の利用が否と設定されている場合には前記探索メッセージにより発見されたネットワークデバイスに対応するデバイスドライバの自動インストールを行わないことを特徴とする情報処理装置の制御方法。

【請求項 8】

ネットワーク通信手段を介してネットワークデバイスを検出し、当該検出したネットワークデバイスに対応するデバイスドライバを自動インストールするコンピュータである情報処理装置が実行するコンピュータプログラムであって、

前記コンピュータを、

接続するネットワークを特定するネットワーク環境情報と当該ネットワーク環境情報で示されるネットワークに対して前記デバイスドライバの自動インストールのための機能の利用の可否を設定する設定手段と、

前記設定手段により設定された情報を記述した設定情報を記憶手段に格納する手段と、

接続中のネットワークに関するネットワーク環境情報を取得する取得手段と、

前記記憶手段により記憶されている設定情報の中から、前記取得手段で取得されたネットワーク環境情報に対応する設定情報を特定する特定手段と、

前記特定手段により特定された設定情報の内容を表示する表示手段と、

前記特定手段により特定された設定情報に従い、接続中のネットワークのネットワークデバイスに対応するデバイスドライバの自動インストールの制御を行う制御手段として機能させ、

前記デバイスドライバの自動インストールのための機能には、接続中のネットワーク上のネットワークデバイスから発行されたネットワーク参加メッセージを前記情報処理装置が搭載しているインストールの制御のためのシステムへ通知する第 1 機能と、ネットワークデバイスの探索メッセージをネットワークへ送信する第 2 機能とが含まれ、

前記設定手段は、ネットワーク環境情報に応じて前記第 1 機能及び前記第 2 機能のそれぞれの利用の可否を設定し、

前記制御手段は、前記設定情報において、前記第 1 機能の利用が可と設定されている場合には前記ネットワーク参加メッセージにより特定されるネットワークデバイスに対応するデバイスドライバの自動インストールを行い、前記第 1 機能の利用が否と設定されている場合には前記ネットワーク参加メッセージにより特定されるネットワークデバイスに対応するデバイスドライバの自動インストールを行わず、前記第 2 機能の利用が可と設定されている場合には前記探索メッセージにより発見されたネットワークデバイスに対応するデバイスドライバの自動インストールを行い、前記第 2 機能の利用が否と設定されている場合には前記探索メッセージにより発見されたネットワークデバイスに対応するデバイスドライバの自動インストールを行わないことを特徴とするコンピュータプログラム。

【請求項 9】

請求項 8 に記載のコンピュータプログラムを格納したことを特徴とするコンピュータ可読記憶媒体。

【発明の詳細な説明】

【技術分野】

## 【 0 0 0 1 】

本発明は、ネットワークに対応した周辺装置（デバイス）を制御するためのデバイスドライバのインストール技術に関するものである。

## 【背景技術】

## 【 0 0 0 2 】

従来から、ネットワーク上のクライアント装置からのサービス要求に応えるサービス提供装置及び、サービス提供システムが知られている。また、インターネットによる通信の飛躍的な普及に伴い、ネットワーク対応型機器として、従来のパーソナルコンピュータ以外の種類の多様な機器が開発されている。例えば、PDA（Personal Digital Assistance）、携帯電話等のユーザインタラクティブなデバイスは勿論こと、スキャナ、プリンタ、複写機、デジタルカメラ等の画像処理装置が挙げられる。また、更には、テレビ、エアコン、冷蔵庫等の家電製品などまでもが、ネットワーク対応になってきている。

10

## 【 0 0 0 3 】

これに伴い、サービスを提供するネットワークデバイスの探索・検索する技術が開発されている。特に、ネットワーク対応型デバイスを制御するためのアプリケーションソフトウェア、ユーティリティソフトウェア、オペレーティングシステム等の自動セットアップを行ない技術、プロトコル、アーキテクチャが提案されている（特許文献1，2）。

## 【 0 0 0 4 】

また、複数の企業、標準化団体が、ローカルI/O接続デバイスに対して適用されてきたプラグアンドプレイをネットワークデバイスに対して拡張対応すべく、仕様策定作業を進めている。

20

## 【 0 0 0 5 】

例えば、米国Microsoft社が主体となって策定を進めるUPnP（登録商標）、およびWSD（Web Services for Devices）、ジネス機械・情報システム産業協会（JBMA）が推進するBMLinks、米国Apple社が開発したOS XでサポートされるRenegadeなどなどがその一例である。

## 【 0 0 0 6 】

上記のように、ネットワークプラグアンドプレイが普及することによって利便性が向上することが期待されている。

【特許文献1】特開2004-038956公報

30

【特許文献2】特開2004-362594公報

## 【発明の開示】

## 【発明が解決しようとする課題】

## 【 0 0 0 7 】

しかしながら、ノートPC、PDA、携帯電話のように、異なるネットワーク環境を常時移動するようなモバイルデバイスを利用する場合、上記のネットワークプラグアンドプレイデバイスによる弊害が予想される。なぜなら、モバイルデバイスを利用する際に、ユーザの意図に反した不必要なネットワークデバイスが検索されてしまい、そのドライバ、アプリケーションがモバイルデバイスにインストールされてしまうからである。

40

## 【 0 0 0 8 】

例えば、ノートパソコンを自宅、あるいは勤務先のオフィスから移動し、訪問先オフィス、あるいは公共施設に設けられたホットスポットなど、異なるネットワーク環境で利用するケースについて考察すると分かりやすい。この場合、ユーザー自身は利用する意図がなくても、ノートパソコンのプラグアンドプレイ機能が、そのネットワーク環境で稼働しているネットワークデバイスを発見してしまうことが起こりえる。すなわち、必要としないネットワークデバイスのドライバ、アプリケーションが自動的にインストールされてしまう事態が起こる。

## 【 0 0 0 9 】

この結果、モバイルデバイスのハードウェア資源、特にメモリ資源をいたずらに消費してしまう。また、セキュリティ上信頼できないソフトウェアがユーザが意識しないうちに

50

インストールされてしまい、情報の破壊、漏洩といった深刻な被害につながることもなろう。

【 0 0 1 0 】

また、同様に、ノート P C、P D A、携帯電話のようなモバイルデバイスが備えるプラグアンドプレイ機能が、ネットワークデバイスを発見するたびに、インストールを促すポップアップメッセージ、あるいは警告メッセージ等を表示することになる。従って、ネットワークデバイスの総数が数十台から数百台に及ぶ場合、モバイルデバイスのユーザは、それらメッセージをキャンセルする手続き、作業に煩わされることになる。

【 0 0 1 1 】

本発明はかかる問題点に鑑みなされたものであり、接続中のネットワークのネットワーク環境に応じて、発見されたネットワークデバイスのデバイスドライバの自動インストールを行う / 行わないを設定可能とする技術を提供しようとするものである。

【課題を解決するための手段】

【 0 0 1 2 】

この課題を解決するため、例えば本発明の情報処理装置は以下の構成を備える。すなわち、

ネットワーク通信手段を介してネットワークデバイスを検出し、当該検出したネットワークデバイスに対応するデバイスドライバを自動インストールする情報処理装置であって、

接続するネットワークを特定するネットワーク環境情報と当該ネットワーク環境情報で示されるネットワークに対して前記デバイスドライバの自動インストールのための機能の利用の可否を設定する設定手段と、

前記設定手段により設定された情報を記述した設定情報を記憶する記憶手段と、

接続中のネットワークに関するネットワーク環境情報を取得する取得手段と、

前記記憶手段により記憶されている設定情報の中から、前記取得手段で取得されたネットワーク環境情報に対応する設定情報を特定する特定手段と、

前記特定手段により特定された設定情報の内容を表示する表示手段と、

前記特定手段により特定された設定情報に従い、接続中のネットワークのネットワークデバイスに対応するデバイスドライバの自動インストールの制御を行う制御手段とを備え、

前記デバイスドライバの自動インストールのための機能には、接続中のネットワーク上のネットワークデバイスから発行されたネットワーク参加メッセージを前記情報処理装置が搭載しているインストールの制御のためのシステムへ通知する第 1 機能と、ネットワークデバイスの探索メッセージをネットワークへ送信する第 2 機能とが含まれ、

前記設定手段は、ネットワーク環境情報に応じて前記第 1 機能及び前記第 2 機能のそれぞれの利用の可否を設定し、

前記制御手段は、前記設定情報において、前記第 1 機能の利用が可と設定されている場合には前記ネットワーク参加メッセージにより特定されるネットワークデバイスに対応するデバイスドライバの自動インストールを行い、前記第 1 機能の利用が否と設定されている場合には前記ネットワーク参加メッセージにより特定されるネットワークデバイスに対応するデバイスドライバの自動インストールを行わず、前記第 2 機能の利用が可と設定されている場合には前記探索メッセージにより発見されたネットワークデバイスに対応するデバイスドライバの自動インストールを行い、前記第 2 機能の利用が否と設定されている場合には前記探索メッセージにより発見されたネットワークデバイスに対応するデバイスドライバの自動インストールを行わないことを特徴とする。

【発明の効果】

【 0 0 1 3 】

本発明によれば、接続中のネットワークのネットワーク環境に応じて、発見されたネットワークデバイスのデバイスドライバの自動インストールを行う / 行わないを設定可能となる。その結果、或るネットワークに接続した際に、ユーザが利用する情報処理装置のメ

10

20

30

40

50

メモリ等のハードウェア資源が不必要に消費されることを防止できる。また、セキュリティ上信頼できないソフトウェアによる情報の破壊、漏洩といった被害を防止することも可能になる。

【発明を実施するための最良の形態】

【0014】

以下、添付図面に従って本発明に係る実施形態を詳細に説明する。なお、以下の説明におけるプロトコル、バージョン、アドレス、その他の数値等は一例であって、特に特定の記載がない限りは、本発明を限定するものとはならないことに注意されたい。

【0015】

<第1の実施形態>

図1は第1の実施形態におけるネットワークプラグアンドプレイシステムの構成を示すブロック図である。

【0016】

図中、100は、クライアントデバイスで、オフィスからホームネットワーク、あるいは訪問先のネットワーク環境へと接続環境が頻繁に変化する可能性の大きなクライアント端末としてのノートPCであるものとする。勿論、PDA、携帯電話等でも構わない。

【0017】

クライアント端末100は、通信機能としてEthernet(登録商標)、Wi-Fi(IEEE80.11a/b/g)、Bluetoothに対応しており、それぞれEthernet(登録商標)コントローラ1、Wi-Fiコントローラ2、Bluetoothコントローラ3を備えるか、それ専用のカードとして搭載可能とする。これらコントローラの上位レイヤには、TCP/UDP/IPプロトコルスタック4を備え、そのプロトコルスタック上にHTTP5を備え、HTTPリクエストの解析、およびレスポンス処理を行う。

【0018】

TCP/UDP/IPプロトコルスタック4、およびHTTP5の上位層にはSOAP(Simple Object Access Protocol)プロセッサ6を備え、プラグアンドプレイモード設定ユーティリティ7、WSDモジュール8、およびユーティリティ14、アプリケーション15が、それぞれ該処理部を介してeXtensible Markup Language(XML)で記述されたデータによる双方向通信を実現する。

【0019】

ネットワークマネージャ9は、各Ethernet(登録商標)コントローラ1、Wi-Fiコントローラ2、Bluetoothコントローラ3を管理するモジュールであり、各コントローラの設定情報、通信状態に関する情報を取得する機能を備える。

【0020】

プラグアンドプレイモード設定ユーティリティ7はネットワークマネージャ9を制御し、現在稼働中のネットワークコントローラ情報、およびその設定情報を取得し、その設定情報をメモリコントローラ10を介して、メモリ(ハードディスクでも構わない)上に記録する機能を有する。

【0021】

また、プラグアンドプレイモード設定ユーティリティ7はTCP/UDP/IPプロトコルスタック4よりネットワーク構成情報を取得する機能を備え、その設定情報をモード設定情報12としてメモリコントローラ10を介して、メモリ(ハードディスク等の不揮発性メモリ)上に記録する機能を有する。

【0022】

また、プラグアンドプレイモード設定ユーティリティ7は、プラグアンドプレイ設定のUIを制御し、その設定内容をメモリコントローラ10をメモリ上に記録する。

【0023】

これら設定情報は、プラグアンドプレイモード設定ユーティリティ7によりXML(eXtensible Markup Language)に変換され、XMLの形式でメモリ上に記録される。設定情報の詳細、および書式に関しては後述する。

10

20

30

40

50

## 【 0 0 2 4 】

プラグアンドプレイモード設定ユーティリティ 7 は、記録されたプラグアンドプレイモードに従い、R C P / U D P / I P プロトコルスタック 4 に対し、H e l l o メッセージ受信ポートのEnable/Disableを制御する。

## 【 0 0 2 5 】

また、プラグアンドプレイモード設定ユーティリティ 7 は、記録されたプラグアンドプレイモードに従い、S O A P プロセッサ 6 に対し、Probeメッセージ送信Enable/Disableを制御する。

## 【 0 0 2 6 】

W S D モジュール 8 は S O A P プロセッサ 6 を介して、米国Microsoft社らが、その仕様策定を推進するW S - D i s c o v e r y 仕様にに基づき、ネットワークデバイスから通知されるH e l l o メッセージに対する応答処理、および、ネットワークデバイス検索のためのP r o b e メッセージの発行処理を実行する。また、W S - M e t a d a t a E x c h a n g e 仕様にに基づき、G e t M e t a d a t a メッセージを発行することで、ネットワークデバイスの属性情報を取得する。

10

## 【 0 0 2 7 】

これらメッセージ処理により、ネットワークデバイスを発見した場合、W S D モジュール 8 はプラグアンドプレイコントローラ 1 1 に対して、発見したネットワークデバイスの属性情報を通知する。プラグアンドプレイコントローラ 1 1 は該属性情報をもとに該当するドライバ、ユーティリティソフトウェアをメモリコントローラ 1 0 を介してメモリより読み込み、クライアント端末 1 0 0 に対してインストールを実行する機能を備える。

20

## 【 0 0 2 8 】

アプリケーションは例えばワードプロセッサなどで編集した書類をネットワークデバイス・プリンタに印刷する際、プラグアンドプレイコントローラ 1 1 によりインストールされたドライバ、ユーティリティを介して、ネットワーク対応デバイス 2 0 0 に対してジョブデータ、印刷データを送信する。

## 【 0 0 2 9 】

一方、ネットワーク対応デバイス 2 0 0 ( 本実施の形態ではネットワーク対応型プリンタとしている ) は、通信機能としてEthernet(登録商標)、Wi-Fi(IEEE80.11a/b/g)、Bluetoothに対応しており、それぞれEthernet(登録商標)コントローラ 1 6、Wi-Fiコントローラ 1 7 により制御される。これらコントローラの上位レイヤには、T C P / U D P / I P プロトコルスタック 1 8 を備え、そのプロトコルスタック上にH T T P 1 9 を備え、H T T P リクエストの解析、およびレスポンス処理を行う。

30

## 【 0 0 3 0 】

T C P / U D P / I P プロトコルスタック 1 8、およびH T T P 1 9 の上位層にはS O A P プロセッサ 2 0 を備え、W S D モジュール 2 1、およびプリンタコントローラ 2 2 が、それぞれ該処理部を介してX M L で記述されたデータの双方向通信を実現する。

## 【 0 0 3 1 】

ここで、W S D モジュール 2 1 はS O A P プロセッサ 2 0 を介して、米国Microsoft社らが、その仕様策定を推進するW S - D i s c o v e r y 仕様にに基づき、ネットワークに接続した際に、H e l l o メッセージの送信、およびクライアント端末 1 0 0 から発行されるP r o b e メッセージに対する応答処理を実行する。

40

## 【 0 0 3 2 】

また、W S - M e t a d a t a E x c h a n g e 仕様にに基づき、クライアント端末 1 0 0 から発行されたGetMetadataメッセージに応じ、ネットワーク対応デバイス(本例では、ネットワーク対応プリンタ)が持つ属性情報を返信する。

## 【 0 0 3 3 】

ワイヤレスL A N アクセスポイント 2 3 は、ネットワーク上の通信の中継装置として機能する。クライアント端末 1 0 0 がW i - F i コントローラ 2 を介してネットワーク接続する際にも、このワイヤレスL A N アクセスポイント 2 3 を介してEthernet(登録商標)

50

に接続することになる。

【 0 0 3 4 】

ここで、クライアント端末 1 0 0 とネットワーク対応デバイス 2 0 0 間の W S D の仕様に基づく探索及びそのネットワーク対応デバイス 2 0 0 用のデバイスドライバのインストールの手順を図 1 9 に従って簡単に説明する。

【 0 0 3 5 】

図 1 9 ( a ) は、クライアント端末 1 0 0 がネットワークに接続する際、もしくはユーザによるネットワークデバイスの探索の指示入力となされた場合の手順を示している。

【 0 0 3 6 】

クライアント端末 1 0 0 は、ネットワークにデバイス検索要求である P r o b e メッセージをマルチキャスト（ブロードキャスト）送信する。この P r o b e メッセージを受信したネットワーク対応デバイス（実施形態でネットワーク対応プリンタ）2 0 0 は、それに応じるべくクライアント端末 1 0 0 に向けて応答メッセージ（P r o b e M a t c h ）をユニキャスト送信する。これ以降、通信相手先が互いに判明しているので、送受信はユニキャストになる。

【 0 0 3 7 】

クライアント端末 1 0 0 は、上記の応答メッセージを受信することで、ネットワーク上にネットワーク対応デバイス 2 0 0 の存在することが検出できる。従って、次の段階に進み、そのネットワーク対応デバイスに対してデバイス情報要求メッセージ（G e t M e t a D a t a ）を送信（ユニキャスト）し、デバイス情報（M e t a D a t a f o r D e v i c e ）を取得する。この後、クライアント端末 1 0 0 は、サービス情報要求メッセージ（G e t M a t a D a t a ）を送信し、サービス情報（M e t a D a t a f o r S e r v i c e ）を取得する。

【 0 0 3 8 】

以上の結果、クライアント端末 1 0 0 は、ネットワーク対応デバイス 2 0 0 の機種名、属性情報（プリンタであることを示す情報）を得ることができるので、O S に該当するデバイスドライバ（上記の場合にはプリンタドライバ）のインストールが行われることになる。

【 0 0 3 9 】

図 1 9 ( b ) は、クライアント端末 1 0 0 は既にネットワークに接続が完了した後、ネットワーク対応デバイス 2 0 0 をネットワークに参加させた場合の手順を示している。なお、ネットワーク対応デバイス 2 0 0 のネットワークへの参加とは、そのネットワーク対応デバイス 2 0 0 の電源が投入された場合と考えると分かりやすい。

【 0 0 4 0 】

まず、ネットワーク対応デバイスはネットワークに参加するための参加メッセージ（H e l l o ）をマルチキャスト送信する。クライアント端末 1 0 0 は、この参加メッセージを受信することで、ネットワーク対応デバイス 2 0 0 の存在を検出するので、デバイス情報要求メッセージを送信する。この後の手順は、図 1 9 ( a ) と同じである。

【 0 0 4 1 】

さて、上記のようにして、クライアント端末 1 0 0 が未知のネットワーク対応デバイスを検出して、デバイスドライバが自動インストールされるので、ユーザにかかる負担は少なくなる。しかしながら、クライアント端末 1 0 0 を自宅等の小規模の L A N 環境から、ネットワーク対応デバイスが多数存在する社内 L A N 環境に移動させると、図 1 9 ( a )、( b ) のいずれの場合においても、多数のデバイスが検索されてしまい、結局のところ、多数のデバイスドライバが次々とインストールされてしまう。また、ネットワーク対応デバイスがネットワークスキャナ、ネットワークプリンタとして機能する複合機である場合には、それぞれの機能に関するドライバがインストールされることになる。更には、ネットワークプリンタ 1 つを例にしても、多数の P D L （ページ記述言語）に対応する場合には、その P D L の数分のデバイスドライバがインストールされることになり、インストールされるデバイスドライバの数は、もはや無視できない数になり得る。1 つのデバイス

10

20

30

40

50

ドライバのインストールに要する時間も数十秒乃至数分になるわけであるから、クライアント端末100を利用した本来の作業を行えるまでに多くの時間を必要とする。なにより不必要なデバイスドライバのためにメモリが消費され、本来の作業に支障を来すことにもなりかねない。

#### 【0042】

そこで、本実施形態では、クライアント端末100がデバイス探索要求メッセージ(Probe)の送信する/しない、ネットワーク対応デバイスのネットワーク参加メッセージ>Hello)に回答する/しないを、そのクライアント端末100が接続するネットワーク環境に応じて設定することで、上記の問題を解決する。

#### 【0043】

以下に本実施形態のクライアント端末100の制御の流れを図2のフローチャートに従い説明する。この処理は、クライアント端末のインストールしたメモリ常駐型のプラグアンドプレイモード設定ユーティリティ(アプリケーションプログラム)の処理でもある。

#### 【0044】

クライアントデバイス100のメモリに常駐しているプラグアンドプレイモード設定ユーティリティ7は、Ethernet(登録商標)、Wi-Fi、Bluetoothのいずれかの通信コントローラを介してネットワーク接続が開始し、ネットワークマネージャ9よりプラグアンドプレイモード設定ユーティリティ7に対しNotifyが通知されるのを待つ(ステップS1)。この通知を受けることをトリガにして、プラグアンドプレイモード設定ユーティリティ7は、ネットワークマネージャ9を介して現在稼働中のネットワークコントローラが、Ethernet、Wi-Fi、Bluetooth Controllerのいずれであるか情報を取得する(ステップS2)。

#### 【0045】

次にプラグアンドプレイモード設定ユーティリティ7はTCP/UDP/IPプロトコルスタック4より、以下のネットワーク設定情報を取得する(ステップS3)。

DHCP Enabled: YES/NO (DHCPによるアドレス取得か否かを意味する)。

IP Address: IPアドレス(ローカルIPアドレス)

Subnet Mask: サブネットマスク値

Default Gateway: Gatewayアドレス

DNS Server: DNSサーバーアドレス

DHCP Server: DHCPサーバーアドレス

#### 【0046】

プラグアンドプレイモード設定ユーティリティ7は、次に取得したデータをメモリコントローラ10を介して、メモリ(ハードディスク)上に、過去に記録されたプラグアンドプレイ設定データファイル(特定の拡張子を持つか、特定のフォルダに格納されているものとする)を検索し(ステップS4)、現在接続されているネットワークの設定情報と一致するプラグアンドプレイ設定データファイルがあるか否かを判断する(ステップS5)。なお、クライアント端末100が過去の一度もネットワークに説明していない場合には、該当するプラグアンドプレイ設定データが存在しないので、不一致と判断する。

#### 【0047】

図3は記録されるプラグアンドプレイ設定データ(ファイル)の構成を示すものであり、ネットワーク情報、およびそのネットワーク環境に対するプラグアンドプレイ設定データであるProbe発行の許可、禁止、Helloメッセージに対する応答許可、禁止の設定情報がXML(eXtensible Markup Language)で記述される。なお、タグ<DHCPEnabled>~</DHCPEnabled>で挟まれた記述が「YES」である場合には、タグ<IPAddress>~</IPAddress>の行は存在しない。DHCPを利用する場合には、クライアント端末100のローカルIPアドレスは、接続する度に变化する可能性があるためである。

#### 【0048】

また、図3のプラグアンドプレイ設定データファイルにおいて、タグ<Probe>~</Probe>は、クライアント端末100からデバイス探索要求メッセージ(Probe

10

20

30

40

50

b e ) をマルチキャスト送信する / しないを記述するものである。また、タグ < H e l l o > ~ < / H e l l o > は、ネットワークデバイスが発行するネットワーク参加メッセージ ( H e l l o ) に対して応答するか否かを記述するものである。

【 0 0 4 9 】

図 1 8 は、既に記憶されたプラグアンドプレイ設定データファイルとの比較手順を示すフローチャートである。

【 0 0 5 0 】

プラグアンドプレイモード設定ユーティリティ 7 は、メモリコントローラ 1 0 を介して順次プラグアンドプレイ設定データファイルを読み込む処理を行う ( ステップ S 1 0 0 ) 。次のステップ S 1 0 1 では、読み込むべきデータが存在し、正常に読み込まれたか否かを判断する。

10

【 0 0 5 1 】

データの読み込みが行われた場合には、ステップ S 1 0 2 に進み、読み込んだデータのタグ < D H C P E n a b l e d > ~ < D H C P E n a b l e d > の記述が Y E S 、 N O のいずれであるかを判断する。つまり、読み込んだデータのネットワーク接続環境が D H C P を利用することが設定されているか否かを判断する。

【 0 0 5 2 】

D H C P を利用する記述がなされていると判断した場合には、ステップ S 1 0 4 に進んで、現在ネットワークに接続中のネットワークコントローラが D H C P クライアントとして設定されているか否かを判断する。つまり、読み込んだデータ中の D H C P に関する記述が、現在のネットワーク接続中の D H C P の状態と同じであるか否かを判断する。否の場合には、読み込んだプラグアンドプレイ設定データファイルは、現在のネットワーク接続環境とは異なるものであるため、ステップ S 1 0 0 に戻り、次のプラグアンドプレイ設定データファイルの読み込みを行う。

20

【 0 0 5 3 】

また、ステップ S 1 0 4 にて、読み込んだプラグアンドプレイ設定データファイル中の D H C P に関する記述と、現在のネットワーク接続環境とが一致していると判断した場合には、ステップ S 1 0 5 に進んで、読み込んだプラグアンドプレイ設定データファイル中の D N S サーバ、 D e f o u l t G a t e w a y の I P アドレスが、現在のネットワーク接続環境におけるそれらの I P アドレスと一致するか否かを判断する。不一致の場合には、ステップ S 1 0 0 に戻り、一致する場合にはステップ S 1 0 6 に進む。

30

【 0 0 5 4 】

ステップ S 1 0 6 では、読み込んだプラグアンドプレイ設定データファイル中の S u b n e t M a s k の情報と、現在のネットワーク接続環境における S u b n e t M a s k が一致しているか否かを判断する。否の場合には、ステップ S 1 0 0 に戻り、一致していればステップ S 1 0 7 に進んで、現在の接続中のネットワーク環境と同じ情報を記述したプラグアンドプレイ設定データファイルが存在すると判定し、本処理を終了する。

【 0 0 5 5 】

また、ステップ S 1 0 2 にて、読み込んだプラグアンドプレイ設定データファイル中の D H C P に関する記述が “ N O ” であった場合には、ステップ S 1 0 3 に進む。ここでは、読み込んだプラグアンドプレイ設定データファイル中のタグ I P a d r e s s に記述された I P アドレスが、現在の接続中のネットワークコントローラに設定された I P アドレスと一致するか否かを判断する。不一致であれば、読み込んだプラグアンドプレイ設定データファイルは、ステップ S 1 0 0 に戻る。また、一致している場合には、ステップ S 1 0 5 に処理を進める。

40

【 0 0 5 6 】

上記のようにして、ステップ S 1 0 0 に戻って上記処理を繰り返し、現在接続中のネットワーク環境に一致するプラグアンドプレイ設定データファイルが見つからない場合には、ステップ S 1 0 1 からステップ S 1 0 8 に進み、一致データファイル無しとして判定し、本処理を終了することになる。

50

## 【 0 0 5 7 】

図2のフローチャートの説明に戻る。上記のようにして、ステップS5において、現在接続中のネットワーク環境と同一の環境を記載したプラグアンドプレイ設定データファイルが存在すると判断した場合には、ステップS6に進んで、そのプラグアンドプレイ設定データファイルを読み込み、クライアント端末100のユーザにその記述内容を変更するか否かを選択させるため、図4に示すポップアップメニューを表示し、変更/変更無しをステップS7にて判断する。

## 【 0 0 5 8 】

図4では、接続されたネットワーク接続環境に対し、ネットワークデバイスの検索を行う(Probeメッセージの送信許可)、ネットワークデバイスの参加メッセージ(Hello)に対して応答しないことが選択されている状態を示している。

10

## 【 0 0 5 9 】

ユーザは表示されたポップアップメニューに対して変更するか否かを選択するため、図示のボタンYes、Noの何れかをポインティングデバイスで指示することになる。

## 【 0 0 6 0 】

変更無し、すなわち、図4のボタンNoが指示されたと判断した場合には、ステップS7からステップS8に進んで、読み込んだプラグアンドプレイ設定データファイルの記述に従ってプラグアンドプレイ制御処理を実行する。

## 【 0 0 6 1 】

一方、ステップS5において、現在接続中のネットワーク環境に一致するプラグアンドプレイ設定データファイルが存在しない、或いは、ステップS7にて変更するとの指示が入力された場合には、処理はステップS9に進み、図5のポップアップメニューを表示し、現在接続中のネットワーク環境に対するプラグアンドプレイモードの設定を利用者に促す。

20

## 【 0 0 6 2 】

このポップアップメニューには、現在クライアント端末100が接続しているネットワーク種別、およびネットワーク情報の一部が表示されており、ネットワークデバイスの検索機能、およびHelloメッセージ応答に対して、それぞれ許可、禁止の設定が可能となっている。

## 【 0 0 6 3 】

ステップS10では、設定が完了したか否かを判断する。図5の画面にて、OKボタンがポインティングデバイスで指示されると設定が完了したと判断し、ステップS10に進んで設定内容を、図3に示すフォーマットにて、メモリコントローラ10を介してメモリ11上に登録し、該設定情報にもとづきプラグアンドプレイ制御を実施する(ステップS8)。

30

## 【 0 0 6 4 】

なお、ステップS5からステップS9に処理が進んだ場合には、プラグアンドプレイ設定データファイルは新規保存となり、ステップS7からステップS9に進んだ場合には、既存のプラグアンドプレイ設定データファイルを変更し、再保存となる。

## 【 0 0 6 5 】

図6は、プラグアンドプレイモード設定に従い、ネットワークデバイスの検索機能、およびHelloメッセージ応答に対して、それぞれ許可、禁止する具体的手段を説明するフローチャートである。

40

## 【 0 0 6 6 】

プラグアンドプレイモード設定ユーティリティ7は、現在、接続中であるネットワークに対するプラグアンドプレイモード設定に基づき、その設定がHello応答禁止になっているか否かを判断する(ステップS21)。Hello応答禁止になっていると判断した場合には、TCP/UDP/IPプロトコルスタック4モジュールに対し、マルチキャストポート(239.255.255.250)の無効化を通知する(ステップS22)。この結果、TCP/UDP/IPプロトコルスタック4モジュールは該ポートをDisableとするため

50

、ネットワークデバイスから通知されるHelloメッセージは、その上位レイヤに位置する各モジュール、およびユーティリティ14、アプリケーション15には通知されない。すなわち、ネットワーク上のデバイスがHelloメッセージを発行したとしても、上位に通知されなくなり、そのデバイスのデバイスドライバのインストール処理に進むことがなくなる。

【0067】

また、Hello応答許可が設定されていると判断した場合には、ステップS23に進んで、TCP/UDP/IPプロトコルスタック4モジュールに対し、マルチキャストポート(239.255.255.250)の有効化を通知する。この結果、TCP/UDP/IPプロトコルスタック4モジュールは該ポートをEnableとするため、ネットワークデバイスから通知されるHelloメッセージは、その上位レイヤに位置する各モジュール、およびユーティリティ14、アプリケーション15に通知される。

10

【0068】

次に処理はステップS24に進む。ここでは、Probe発行禁止設定がされているか否かを判断する。Probe発行禁止が設定されている場合、ステップS25に進み、Add Printer Wizard等、ネットワークデバイスの検索機能を有するユーティリティ14、およびアプリケーション15に対し、該当機能のDisableを通知するように設定する。また、SOAPプロセッサ6に対し、Probeパケットの生成禁止を通知する(ステップS26)。この結果、SOAPプロセッサ6は上位レイヤに位置するモジュール、あるいはユーティリティ14、アプリケーション15からProbeパケット発行要求があった場合、エラーを通知する。従って、クライアント端末100から、ネットワークに対してネットワークデバイス検索メッセージ(Probe)が発行されることは無い。

20

【0069】

また、Probe発行許可の設定がなされている場合には、ステップS27に進み、Add Printer Wizard等、ネットワークデバイスの検索機能を有するユーティリティ14、およびアプリケーション15に対し、該当機能のEnableを通知する。併せてSOAPプロセッサ6に対し、Probeパケットの生成許可を通知する(ステップS28)。SOAPプロセッサ6は上位レイヤに位置するモジュール、あるいはユーティリティ14、アプリケーション15からProbeパケット発行要求に応じ、WS-Discovery仕様で定義されたProbeパケットを生成、TCP/UDP/IPコントローラ4を介してProbeパケットをネットワーク上に送信することになる。

30

【0070】

以上、プラグアンドプレイモード設定に対する制御が完了した後、クライアント端末100はネットワークデバイスに対するプラグアンドプレイ処理を実施する。

【0071】

ここでは、ネットワーク対応デバイスがネットワークに参加する際に発行するHelloメッセージが発行される場合の例を図7のフローチャートに従って簡単に説明する。

【0072】

本実施形態において、ネットワーク対応デバイス200は、ネットワークに新規に導入されてネットワーク上で通信を開始する際、あるいは、電源オフの状態から電源オンされてネットワーク上での通信を再開する際、WSDモジュール21は、SOAPプロセッサ20に対してHelloメッセージの送信を要求する。SOAPプロセッサ20は、WS-Discovery仕様に基づきHelloメッセージを生成し、プロトコルスタックを介してマルチキャストアドレス239.255.255.250に対してマルチキャスト送信する(ステップS31)。

40

【0073】

このHelloメッセージはXML-SOAPで記述されており、一例を示すのであれば図8に示すフォーマットである。

【0074】

クライアント端末100において、プラグアンドプレイモードの設定がHelloメッ

50

セージに対して応答禁止するように設定された場合には、TCP/UDP/IPプロトコルスタック4は、マルチキャストポート239.255.255.250を無効としているため、ネットワークデバイス200から送信されたHelloメッセージは破棄される(ステップS32)。すなわち、受信したHelloメッセージの、OSが有するSOAPプロセッサへの通知を遮断する。

【0075】

従って、ネットワーク上に新規ネットワーク対応デバイスが稼動を開始した場合であっても、TCP/UDP/IPプロトコルスタック4より上位レイヤーに位置するアプリケーション15、ユーティリティ14、およびWSDモジュール8は、ネットワーク対応デバイス200の稼動開始情報を取得することは無く、そのため、例えば、ネットワークデバイスをインストールするためのユーティリティが起動したり、あるいは新規デバイス発見を通知するメッセージがディスプレイ上に表示されることもなくなる。

10

【0076】

一方、Hello応答許可設定の場合は、HelloメッセージはTCP/UDP/IPプロトコルスタック4を介して、SOAPプロセッサ6に通知される。SOAPプロセッサ6は、Helloメッセージの内容を解析し、その結果をWSDモジュール8に通知する(ステップS33)。

【0077】

WSDモジュール8は、Helloメッセージに記述されたアドレスに対し、該メッセージを発行したネットワークデバイスの属性情報を取得するために、SOAPプロセッサ6に対してWS-MetadataExchangeで規定されるGetMetadataメッセージ発行を要求することになる。

20

【0078】

SOAPプロセッサ6は、WS-MetadataExchange仕様にに基づきGetMetadataメッセージを生成し、生成されたメッセージをTCP/UDP/IPプロトコルスタック4を介してネットワーク対応デバイス200に対してユニキャスト送信する(ステップS34)。

【0079】

GetMetadataメッセージはXML-SOAPで記述されており、一例として、図9にそのメッセージフォーマットを示す。

30

【0080】

ネットワーク対応デバイス200が、上記GetMetadataメッセージを受信すると、TCP/UDP/IPプロトコルスタック18を介して、SOAPプロセッサ20にそのメッセージが通知される(ステップS35)。SOAPプロセッサ20は、GetMetadataメッセージの内容を解析し、その結果をWSDモジュール21に通知する。WSDモジュール21は、指定された属性情報をクライアントに対して返信するために、SOAPプロセッサ20に対してWS-MetadataExchangeで規定されるGetMetadataResponseメッセージ発行を要求する。

【0081】

SOAPプロセッサ20は、WS-MetadataExchange仕様にに基づきGetMetadataResponseメッセージを生成し、プロトコルスタックを介してクライアント端末100に対してユニキャスト送信することになる(ステップS36)。

40

【0082】

GetMetadataResponseメッセージはXML-SOAPで記述されており、一例を示すのであれば図10のフォーマットである。

【0083】

クライアント端末100がGetMetadataResponseメッセージを受信すると、そのメッセージがTCP/UDP/IPプロトコルスタック4を介して、SOAPプロセッサ6に通知される(ステップS37)。

50

## 【 0 0 8 4 】

Get Metadata Response メッセージに記述されたデバイスの属性情報は、クライアント端末 100 の SOAP プロセッサ 6 を介して、デバイス属性情報が WSD モジュール 8 に通知される。

## 【 0 0 8 5 】

次に、WSD モジュール 8 は、プラグアンドプレイコントローラ 11 を起動し(ステップ S 3 8)、デバイス属性情報をプラグアンドプレイコントローラ 11 に対し通知する。通知を受けたプラグアンドプレイコントローラ 11 は、メモリコントローラを介してデバイス属性情報に一致するドライバを、自身が有する記憶装置上より検索し、該当するドライバを OS にインストールする処理を実行する(ステップ S 3 9)。

10

## 【 0 0 8 6 】

次に、クライアント端末のネットワーク対応デバイスの検索処理について図 11 のフローチャートに従って簡単に説明する。

## 【 0 0 8 7 】

通常、クライアント端末 100 内のユーティリティ 14、あるいはアプリケーション 15 が、SOAP プロセッサ 6 に対し WS-Discovery 仕様にて規定される Probe メッセージの送信を要求すると、SOAP プロセッサ 6 は、WS-Discovery 仕様に基づき Probe メッセージを生成し、プロトコルスタックを介してマルチキャストアドレス 239.255.255.250 に対してマルチキャスト送信する。そして、そのメッセージに対するネットワーク対応デバイス 200 からの応答である Probe Match メッセージの有無により検索処理を実現している。

20

## 【 0 0 8 8 】

ただし、プラグアンドプレイモードの設定で、Probe 発行禁止が設定されている場合(図 6 のステップ S 2 5、S 2 6)、ユーティリティ 14、アプリケーション 15 は、該当機能が Disable となっているため、SOAP プロセッサ 6 に対して Probe メッセージ発行要求が通知されることは無い。

## 【 0 0 8 9 】

また、先に説明したように、プラグアンドプレイモード設定ユーティリティ 7 より、SOAP プロセッサ 6 に対して Probe パケット生成禁止が通知されているため、SOAP プロセッサ 6 は上位レイヤに位置するモジュール、あるいはアプリケーションから Probe パケット発行要求があった場合、エラーを通知する(ステップ S 4 1)。そのため、クライアント端末 100 から、ネットワークに対してネットワークデバイス検索の Probe メッセージが発行されることは無い。従って、ネットワーク対応デバイスのデバイスドライバのインストールされることも無い。

30

## 【 0 0 9 0 】

一方、プラグアンドプレイモードが、Probe 発行許可に設定されている場合、WS-Discovery 仕様で規定される Probe メッセージが SOAP プロセッサ 6 より、TCP/UDP/IP プロトコルスタック 4 を介してマルチキャストアドレス 239.255.255.250 に対してマルチキャスト送信される(ステップ S 4 2)。

## 【 0 0 9 1 】

Probe メッセージは、XML-SOAP で記述されており、一例を示すのであれば図 12 のメッセージフォーマットである。

40

## 【 0 0 9 2 】

ネットワーク対応デバイス 200 は、この Probe メッセージを受信すると(ステップ S 4 3)、受信した Probe メッセージは TCP/UDP/IP プロトコルスタック 16 を介して、SOAP プロセッサ 20 に通知される。SOAP プロセッサ 20 は、Probe メッセージの内容を解析し、その結果を WSD モジュール 21 に通知する。WSD モジュール 21 は、通知されたメッセージの内容が、このネットワーク対応デバイス 200 が提供する機能と一致しているか否かを判断し(ステップ S 4 4)、一致しないと判断した場合、その情報を無視する。したがって、この場合はネットワーク対応デバイス 20

50

0 から Probe レスポンスが発行されることは無い。

【0093】

一方、一致すると判断した場合、WSDモジュール21においては、SOAPプロセッサ20に対してProbe Matchメッセージ発行を要求する。

【0094】

SOAPプロセッサ6は、WS-Discovery仕様に基つきProbe Matchメッセージを生成し、プロトコルスタックを介してクライアント端末100に対してユニキャスト送信する(ステップS45)。

【0095】

Probe MatchメッセージはXML-SOAPで記述されており、図13はそのメッセージフォーマットの一例である。

10

【0096】

クライアント端末100が、上記のProbe Matchメッセージを受信すると(ステップS46)、TCP/UDP/IPプロトコルスタック4を介して、SOAPプロセッサ6に通知される。SOAPプロセッサ6は、Probe Matchメッセージの内容を解析し、その結果をWSDモジュール8に通知する。

【0097】

WSDモジュール8は、Probe Matchメッセージに記述されたアドレスに対し、該メッセージを発行したネットワーク対応デバイス200の属性情報を取得するために、SOAPプロセッサ6に対してWS-MetadataExchangeで規定されるGetMetadataメッセージ発行を要求する。

20

【0098】

SOAPプロセッサ6は、WS-MetadataExchange仕様に基つきGetMetadataメッセージを生成し、プロトコルスタックを介してネットワーク対応デバイス200に対してユニキャスト送信する(ステップS47)。

【0099】

GetMetadataメッセージはXML-SOAPで記述されており、既に図9で示した通りである。

【0100】

ネットワーク対応デバイス200は、このGetMetadataメッセージを受信すると、TCP/UDP/IPプロトコルスタック18を介して、SOAPプロセッサ20に通知する(ステップS48)。SOAPプロセッサ20は、GetMetadataメッセージの内容を解析し、その結果をWSDモジュール21に通知する。WSDモジュール21は、指定された属性情報をクライアントに対して返信するために、SOAPプロセッサ20に対してWS-MetadataExchangeで規定されるGetMetadataResponseメッセージ発行を要求する。

30

【0101】

SOAPプロセッサ20は、WS-MetadataExchange仕様に基つきGetMetadataResponseメッセージを生成し、プロトコルスタックを介してクライアント端末100に対してユニキャスト送信する(ステップS49)。

40

【0102】

GetMetadataResponseメッセージはXML-SOAPで記述されており、その記載フォーマットは既に説明した図10に示した通りである。

【0103】

クライアント端末100が、このGetMetadataResponseメッセージを受信すると(ステップS51)、受信したメッセージはTCP/UDP/IPプロトコルスタック4を介して、SOAPプロセッサ6に通知される。

【0104】

GetMetadataResponseメッセージに記述されたデバイスの属性情報は、クライアント端末100のSOAPプロセッサ6を介して、デバイス属性情報がWS

50

Dモジュール8に通知される。この結果、WSDモジュール8は、プラグアンドプレイコントローラ11を起動し(ステップS52)、デバイス属性情報をプラグアンドプレイコントローラ11に通知する。通知を受けたプラグアンドプレイコントローラ11はメモリコントローラを介してデバイス属性情報に一致するドライバを、ハードディスク等の記憶装置上より検索し、該当するドライバのインストールを実行する(ステップS53)。

#### 【0105】

以上、説明した各工程を実行することで、クライアント端末100は異なるネットワーク環境に対して、個別にプラグアンドプレイのモードを設定することが可能となり、それぞれのネットワーク環境に適するプラグアンドプレイが実行可能となる。特に、未知にネットワークに接続した際に、無条件にネットワーク対応デバイスのデバイスドライバがイン

10

#### 【0106】

なお、実施形態では、クライアント端末100がネットワークに接続した際に、そのネットワークが過去に接続経験のあるネットワークの場合(図2のステップS5でYesの場合)、設定内容を変更するか否かを確認するためのポップアップメニューを表示したが、表示しないで直ちにプラグアンドプレイ設定データファイルに従った処理を行うようにしても構わない。ただし、このままでは、プラグアンドプレイ設定データファイルを再編集することができなくなるので、別途ユーティリティアプリケーションから編集できるようにしても良い。

20

#### 【0107】

また、上記実施形態では、接続経験のあるネットワークの数分のプラグアンドプレイ設定データファイルが作成されることになるが、これらを1つのファイルで記憶管理するようにしても良い。1つのファイルで管理する場合、そのデータ構造はバイナリ形式になるが、図20のようなものとすることができる。図示では、過去に3つのネットワークに接続したことがある例を示している。なお、図中「...」は、そのデータ項目はネットワーク判定の条件には含めないことを示している。

#### 【0108】

##### <第2の実施形態>

上記第1の実施形態では、プラグアンドプレイ設定データファイルには、ネットワークの通信にかかるパラメータを記述するものであったが、利用する通信コントローラの種類(実施形態では、Ethernet(登録商標)、Wi-Fi、Bluetoothコントローラの3つ)に従ってProbeメッセージ発行許可/不許可、Helloメッセージに対する応答/応答禁止を設定する例を第2の実施形態として説明する。

30

#### 【0109】

図14は、第2の実施形態における、各ネットワークコントローラ毎に、ネットワークプラグアンドプレイモードを設定する手順を示すフローチャートである。

#### 【0110】

プラグアンドプレイモード設定ユーティリティ7を起動する(ステップS61)、ネットワークマネージャ9に対して、現在利用可能な状態にあるネットワークコントローラの情報

40

#### 【0111】

図15は、第2の実施形態におけるプラグアンドプレイモード設定ユーティリティ7のユーザインターフェイスを示すものであるが、ネットワークマネージャより取得した情報は、“ネットワークを指定してください”の指定項目において、プルダウンメニューの形で表示される。実施形態の場合、Ethernet(登録商標)、Wi-Fi、Bluetoothのいずれかを選択することが可能である。そして、選択したネットワーク接続形態に対して、ネットワーク対応デバイス200の検索機能、およびHelloメッセージ応答に対して、それぞれ許可、禁止の設定が可能となっており、いずれもプルダウンメニューにより、許可、禁止のいずれかを選択設定することが可能である(ステップS63)。

50

## 【 0 1 1 2 】

ここでOKボタンをポインティングデバイスで指示すると、設定完了であるものとして判断し（ステップS 6 4）、設定情報を図 1 6 に示すフォーマットにてハードディスク等のメモリ上に記録する（ステップS 6 5）。

## 【 0 1 1 3 】

この設定はクライアントデバイス1 0 0 がネットワークに接続していない状態であっても可能であり、現在使用中のネットワーク環境から、例えば訪問先オフィスにおいてネットワーク利用することが予めわかっている場合などにおいて、事前にプラグアンドプレイモードを設定しておくことが可能となる。

## 【 0 1 1 4 】

図 1 7 は、クライアント端末1 0 0 がネットワーク接続した際のプラグアンドプレイモード設定ユーティリティ7の処理手順を示すフローチャートである。

## 【 0 1 1 5 】

クライアント端末1 0 0 がEthernet（登録商標）、Wi-Fi、Bluetoothのいずれかの通信媒体を介してネットワーク接続が開始されると、ネットワークマネージャ9よりプラグアンドプレイモード設定ユーティリティ7に対しNotifyが通知される（ステップS 7 1）。プラグアンドプレイモード設定ユーティリティ7は、この通知をトリガーとしネットワークマネージャ9を介して現在稼働中のネットワークコントローラが、Ethernet、Wi-Fi、Bluetooth Controllerのいずれであるか情報を取得する（ステップS 7 2）。

## 【 0 1 1 6 】

次に取得したデータをメモリコントローラ1 0を介して、メモリ上に記録されたプラグアンドプレイ設定データの検索と比較処理を行う（ステップS 7 3）。そして、現在稼働中のネットワーク接続しているネットワークコントローラを記述したプラグアンドプレイ設定データが発見できたか否かを判断する（ステップS 7 4）。

## 【 0 1 1 7 】

一致するプラグアンドプレイ設定データが発見できた場合には、ステップS 7 5に進み、そのプラグアンドプレイ設定情報を読み込み、その設定情報を図 4 に示すとおりポップアップメニューを表示し、ユーザに対し、設定を変更するか否かの判断を促す。ここでユーザがNoボタンがポインティングデバイスで指示された場合、読み込んだプラグアンドプレイ設定データに従ってプラグアンドプレイ制御を実施する（ステップS 7 7）。

## 【 0 1 1 8 】

また、図 4 のメニューにてYesボタンが指示された場合、すなわち、プラグアンドプレイ設定を変更する指示が入力された場合はステップS 7 8に進む。また、ステップS 7 4にて、現在ネットワーク接続しているネットワークコントローラの種類を記述したプラグアンドプレイ設定データが存在しなかった場合、即ち事前に利用が想定されていなかったネットワーク環境であり、プラグアンドプレイモード設定が実行されていない場合にもステップS 7 8に進む。ステップS 7 8では、図 5 に示すポップアップメニューを表示、このネットワーク環境に対するプラグアンドプレイモードの設定を利用者に促す。

## 【 0 1 1 9 】

ポップアップメニューには、現在クライアントデバイス1 0 0 が接続しているネットワーク種別、およびネットワーク情報の一部が表示されており、ネットワーク対応デバイス2 0 0の検索機能、およびHelloメッセージ応答に対して、それぞれ許可、禁止の設定が可能となっている。

## 【 0 1 2 0 】

図 5 のメニューにて、OKボタンが指示されると、設定が完了したものとして判断し（ステップS 7 9）、その設定内容を前述の図 1 6 に示すフォーマットにてハードディスク等のメモリ上に保存する（ステップS 8 0）。なお、このステップS 8 0の保存処理であるが、ステップS 7 6からステップS 7 8に処理が進んだ場合には再保存となる。そして、ステップS 7 4からステップS 7 8に処理が進んだ場合には、新規保存となる。この後、設定内容に従って、プラグアンドプレイ制御を実行する（ステップS 7 7）。

10

20

30

40

50

## 【 0 1 2 1 】

この設定に基づく、ネットワーク対応デバイス 200 の検索処理、および Hello メッセージ応答に対するそれぞれ許可、禁止する具体的な処理は、先に説明した第 1 の実施形態と同様であるので、その説明は省略する。

## 【 0 1 2 2 】

## &lt; 第 3 の実施形態 &gt;

上記第 1 の実施形態では、Probe 発行の許可 / 禁止、Hello メッセージに対する応答可 / 不可の判定を、ネットワーク接続環境の DHCP を利用する / しない、ローカル IP アドレス、サブネットマスク、DNS サーバや DHCP サーバの IP アドレス等のネットワーク環境に基づき判定した。また、第 2 の実施形態では、ネットワークコントローラの種類をネットワーク環境として判定した。

10

## 【 0 1 2 3 】

Probe 発行の許可 / 禁止、Hello メッセージに対する応答可 / 不可の判定を、ネットワーク接続環境の DHCP を利用する / しない、ローカル IP アドレス、サブネットマスク、DNS サーバや DHCP サーバの IP アドレス、更には、ネットワークコントローラの種類に基づいて判定するようにしても構わない。

## 【 0 1 2 4 】

## &lt; その他の実施形態 &gt;

上述した各実施形態においては、ネットワークデバイスの検索、およびネットワークデバイスがネットワークに参加したことの通知を、WS-Discovery で定義されるプロトコルを採用しているが、この限りではなく例えば UPnP v1 で規定される SSDP (Simple Service Discovery Protocol)、GENA (General Event Notification Architecture) などを利用することでも実現可能である。また、ネットワークデバイスの検索プロトコルとして、ネットワークデバイスの属性情報を取得するためのアドレス、およびネットワークデバイスを制御するための制御情報送信先が取得することが可能であるプロトコルであれば、いずれのプロトコルであっても適用可能である。従って、本発明は上記実施形態によって限定されるものではない。

20

## 【 0 1 2 5 】

また上述した実施形態においては、通信媒体として Ethernet (登録商標)、Wi-Fi (ワイヤレス通信 IEEE802.11 a/b/g)、および Bluetooth を利用する例を説明したが、デバイスの発見、およびデバイスが機能実行を開始したことを通知することが可能な通信媒体であれば、ローカル I/O、ネットワークのいずれの場合も適用可能である。

30

## 【 0 1 2 6 】

また上述した実施形態においては、プラグアンドプレイの一例として、発見したネットワークデバイスに対応したドライバのインストールを採用しているが、ネットワークデバイスを利用、制御するために必要なユーティリティ、アプリケーションなどをクライアントデバイス 100 に自動的にインストールする場合においても適用可能である。

## 【 0 1 2 7 】

また、上述した実施形態においてネットワーク対応デバイスとしてプリンタを例に挙げたが、これはあくまで例示であり、通信媒体を介して利用、制御可能なデバイスであれば、スキャナ、ストレージデバイス等、いずれのデバイスに対しても適用可能である。

40

## 【 0 1 2 8 】

また、上述した実施形態においては、クライアント端末 100 がネットワークデバイスを制御するためのドライバをメモリ上に保持する形態を示したが、ドライバに限らずアプリケーション、ユーティリティソフトウェアに対しても適用可能であり、また、これらソフトウェアはネットワークデバイス上に保持されている場合、あるいは第 3 のサーバー上に保持されている場合にも適用可能である。

## 【 0 1 2 9 】

また、上述した実施形態においては、ネットワーク環境を識別するための情報として、DHCP Enabled、IP Address、Subnet Mask、Default Gateway、DNS Server、DHCP Server

50

を利用した実施形態を示したが、一義的にネットワーク環境を識別することが可能であれば良いので他の情報でも構わない。

【0130】

また、上述した実施形態におけるクライアント端末100及びネットワーク対応デバイス200における各処理機能は、各処理機能を実現する為のプログラムをメモリから読み出してCPU（中央演算装置）が実行することによりその機能を実現させるものであるが、これに限定されるものではなく、各処理機能の全部または一部の機能を専用のハードウェアにより実現してもよい。また、上述したメモリは、光磁気ディスク装置、フラッシュメモリ等の不揮発性のメモリや、CD-ROM等の読み出しのみが可能な記録媒体、RAM以外の揮発性のメモリ、あるいはこれらの組合せによるコンピュータ読み取り、書き込み可能な記録媒体より構成されてもよい。

10

【0131】

また、上記実施形態で説明したように、本発明の主要な特徴はクライアント端末100におけるアプリケーションプログラムにある。通常、アプリケーションプログラムは、CD-ROM等のコンピュータ可読記憶媒体に記憶されている。そして、そのコンピュータ可読記憶媒体をコンピュータが読み込み可能な読取装置（例えばCD-ROMドライブ等）にセットし、システムにコピーもしくはインストールすることで、実行可能になる。従って、このようなコンピュータ可読記憶媒体も、当然に本発明の範疇になるのは明らかである。

【図面の簡単な説明】

20

【0132】

【図1】実施形態におけるネットワークプラグアンドプレイシステムの概略構成を示すブロック図である。

【図2】実施形態におけるプラグアンドプレイモード情報の取得、および設定手順を示すフローチャートである。

【図3】実施形態におけるプラグアンドプレイ設定データのフォーマットを示す図である。

【図4】実施形態におけるプラグアンドプレイモード設定情報表示ポップアップメニューの一例を示す図である。

【図5】実施形態におけるプラグアンドプレイモード設定用ポップアップメニューの一例を示す図である。

30

【図6】実施形態におけるプラグアンドプレイモード設定に従い、ネットワークデバイスの検索とHelloメッセージ応答に対する処理内容を示すフローチャートである。

【図7】実施形態におけるクライアント端末とネットワークデバイスとの間でのプラグアンドプレイ処理の流れを示すフローチャートである。

【図8】WS-Discovery仕様で定義されるHelloメッセージフォーマットを示す図である。

【図9】WS-MetadataExchange仕様で定義されるGetMetadataメッセージのフォーマットを示す図である。

【図10】WS-MetadataExchange仕様で定義されるGetMetadataResponseメッセージのフォーマットを示す図である。

40

【図11】実施形態におけるクライアント端末とネットワークデバイスとの間でのプラグアンドプレイ処理の流れを示すフローチャートである。

【図12】WS-Discovery仕様で定義されるProbeメッセージのフォーマットを示す図である。

【図13】WS-Discovery仕様で定義されるProbeMatchメッセージフォーマットを示す図である。

【図14】第2の実施形態におけるネットワークコントローラの種類毎のネットワークプラグアンドプレイモードを設定する手順を示すフローチャートである。

【図15】第2の実施形態におけるプラグアンドプレイモード設定ユーティリティのユー

50

ザインターフェイスを示す図である。

【図 16】第 2 の実施形態におけるプラグアンドプレイ設定データのフォーマット示す図である。

【図 17】第 2 の実施形態におけるプラグアンドプレイモード情報の取得、および設定手順を示すフローチャートである。

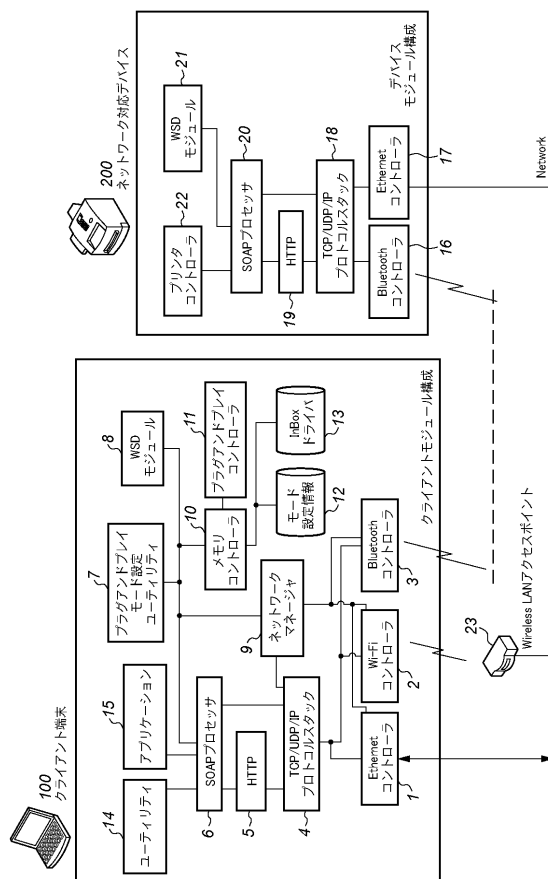
【図 18】第 1 の実施形態におけるネットワーク環境の比較処理手順を示すフローチャートである。

【図 19】U P n P ネットワークデバイスの発見からそのデバイスドライバのインストールされるまでの一般的なシーケンスを示す図である。

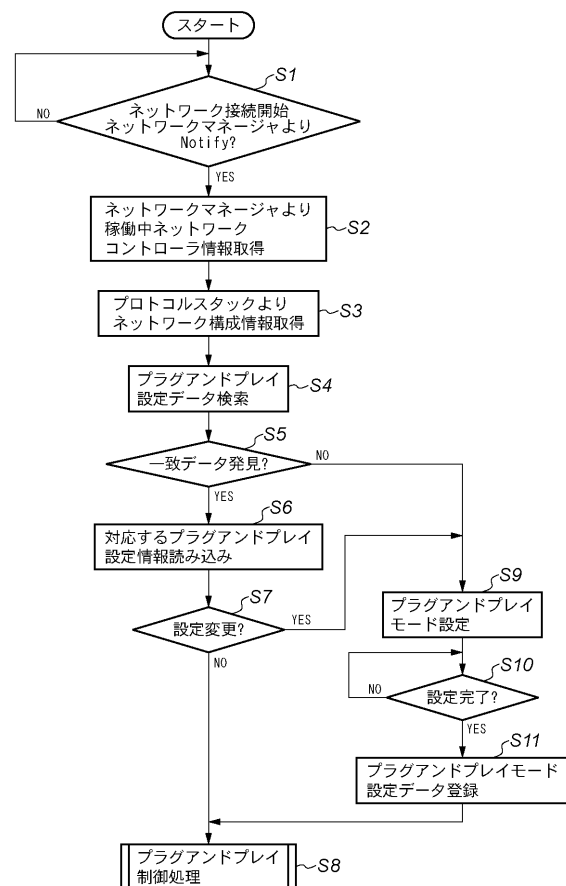
【図 20】第 1 の実施形態におけるプラグアンドプレイ設定データのテーブル形式で保持する場合のデータフォーマットを示す図である。

10

【図 1】



【図 2】



【図 3】

```

<PnPModeData>

  <NetConfiguration>
    <DHCPEnabled>YES|NO</DHCPEnabled>
    <IPAddress>xxx.xxx.xxx.xxx</IPAddress>
    <SubnetMask>xxx.xxx.xxx.xxx</SubnetMask>
    <Gateway>xxx.xxx.xxx.xxx</Gateway>
    <DNSServer>xxx.xxx.xxx.xxx</DNSServer>
    <DHCPSTServer>xxx.xxx.xxx.xxx</DHCPSTServer>
  </NetConfiguration>
  <PnPSetting>
    <Probr>YES|NO</Probe>
    <Hello>YES|NO</Hello>
  </PnPSetting>
</PnPModeData>

```

【図 4】

現在 Ethernet に接続しています

◎ ネットワークデバイスの検索 : 許可

◎ Helloメッセージの応答 : 禁止

設定を変更しますか？

Yes No

【図 5】

現在 Ethernet に接続しています

◎ ネットワークデバイスの検索

許可 ▼

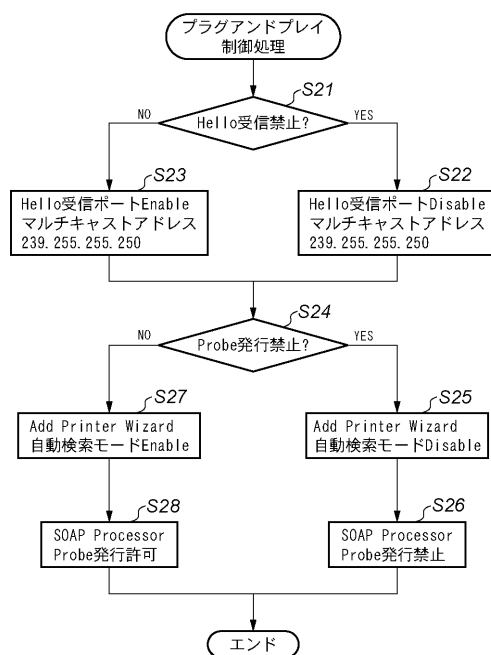
禁止

◎ Helloメッセージの応答

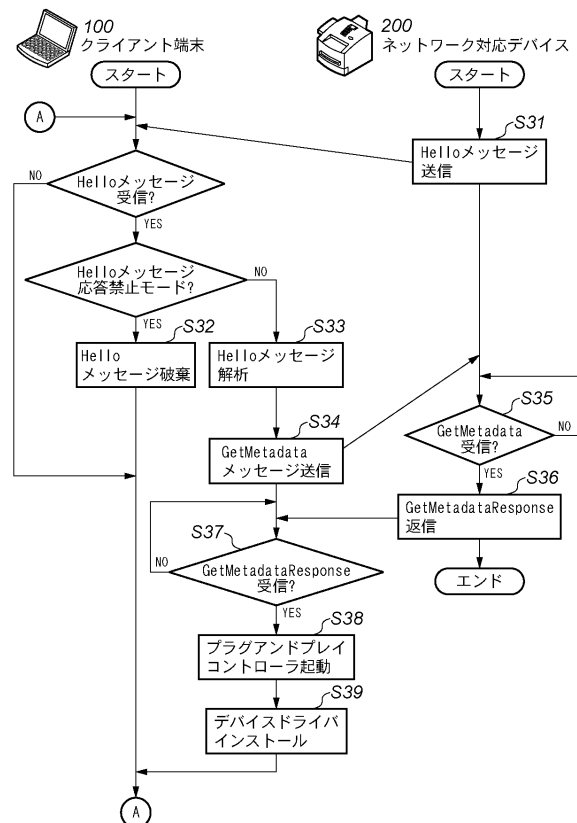
禁止 ▼

OK Cancel

【図 6】



【図 7】



【図 8】

```

<s:Envelope ... >
  <s:Header ... >
    <a:Action ... >
      http://schemas.xmlsoap.org/ws/2005/04/discovery/Hello
    </a:Action>
    <a:MessageID ... >xs:anyURI</a:MessageID>
    [<a:RelatesTo RelationshipType="d:Suppression" >
      xs:anyURI
    </a:RelatesTo>]?
    <a:To ... >urn:schemas-xmlsoap-org:ws:2005:04:discovery</a:To>
    <d:AppSequence ... />
    ...
  </s:Header>
  <s:Body ... >
    <d:Hello ... >
      <a:EndpointReference ... </a:EndpointReference>
      [<d:Types>list of xs:QName</d:Types>]?
      [<d:Scopes>list of xs:anyURI</d:Scopes>]?
      [<d:XAddrs>list of xs:anyURI</d:XAddrs>]?
      <d:MetadataVersion>xs:unsignedInt</d:MetadataVersion>
      ...
    </d:Hello>
  </s:Body>
</s:Envelope>

```

【図 9】

```

<s12:Envelope
  xmlns:s12=' http://www.w3.org/2003/05/soap-envelope'
  xmlns:wsa=' http://schemas.xmlsoap.org/ws/2004/08/addressing'
  xmlns:wsp=' http://schemas.xmlsoap.org/ws/2004/09/policy'
  <s12:Header>
    <wsa:Action>
      http://schemas.xmlsoap.org/ws/2004/09/mex/GetMetadata/Request
    </wsa:Action>
    <wsa:MessageID>
      uuid:73d7edfc-5c3c-49b9-ba46-2480caee43e9
    </wsa:MessageID>
    <wsa:ReplyTo>
      <wsa:Address>http://client.example.com/MyEndpoint</wsa:Address>
    </wsa:ReplyTo>
    <wsa:To>http://server.example.org/YourEndpoint</wsa:To>
    <ex:MyRefProp xmlns:ex=' http://server.example.org/refs' >
      78f2dc229597b529b81c4bef76453c96
    </ex:MyRefProp>
  </s12:Header>
  <s12:Body>
    <wsx:GetMetadata>
      <wsx:Dialect>
        http://aaa.bbb.com/printer/DeviceData
      </wsx:Dialect>
    </wsx:GetMetadata>
  </s12:Body>
</s12:Envelope>

```

【図 10】

```

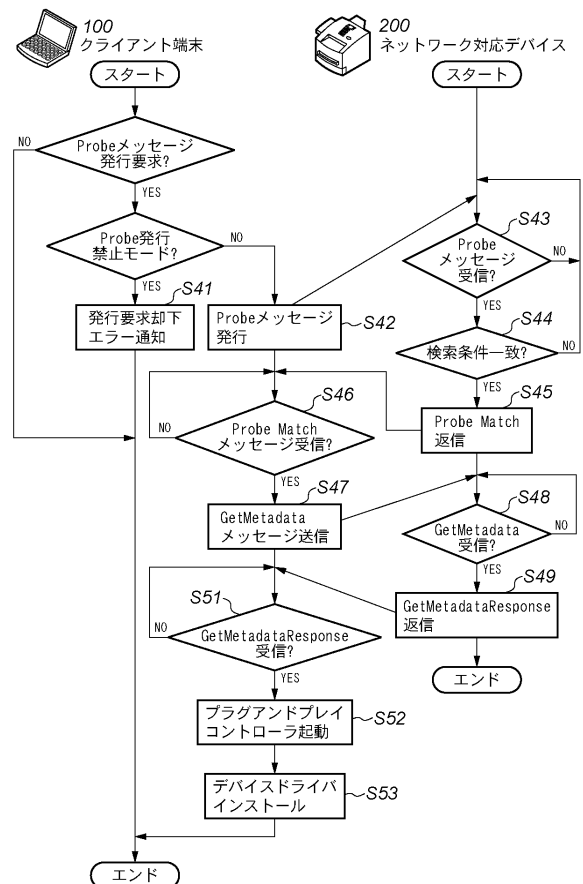
<s12:Envelope
  xmlns:s12=' http://www.w3.org/2003/05/soap-envelope'
  xmlns:wsa=' http://schemas.xmlsoap.org/ws/2004/08/addressing'
  xmlns:wsp=' http://schemas.xmlsoap.org/ws/2004/09/policy'
  xmlns:wsx=' http://schemas.xmlsoap.org/ws/2004/09/mex' >
  <s12:Header>
    <wsa:Action>
      http://schemas.xmlsoap.org/ws/2004/09/mex/GetMetadata/Response
    </wsa:Action>
    <wsa:RelatesTo>
      uuid:73d7edfc-5c3c-49b9-ba46-2480caee43e9
    </wsa:RelatesTo>
    <wsa:To>http://client.example.com/MyEndpoint</wsa:To>
  </s12:Header>
  <s12:Body>
    <wsx:Metadata>
      <wsx:MetadataSection>

        Dialect=' http://aaa.bbb.com/printer/DeviceData'
        <DeviceData>
          <Manufacturer>xxx company</Manufacturer>
          <ModelName>NetworkPrinter123</ModelName>
        </DeviceID>

      </wsx:MetadataSection>
    </wsx:Metadata>
  </s12:Body>
</s12:Envelope>

```

【図 11】



【図 12】

```

<s:Envelope
  xmlns:a="http://schemas.xmlsoap.org/ws/2004/08/addressing"
  xmlns:d="http://schemas.xmlsoap.org/ws/2005/04/discovery"
  xmlns:i="http://printer.example.org/ws/2003/imaging"
  xmlns:s="http://www.w3.org/2003/05/soap-envelope">
  <s:Header>
    <a:Action>
      http://schemas.xmlsoap.org/ws/2005/04/discovery/Probe
    </a:Action>
    <a:MessageID>
      uuid:0a6dc791-2be6-4991-9af1-454778a1917a
    </a:MessageID>
    <a:To>urn:schemas-xmlsoap-org:ws:2005:04:discovery</a:To>
  </s:Header>
  <s:Body>
    <d:Probe>
      <d:Types>i:PrintBasic</d:Types>
      <d:Scopes>
        MatchBy="http://schemas.xmlsoap.org/ws/2005/04/discovery/ldap"
        ldap:///ou=engineering,o=examplecom,c=us
      </d:Scopes>
    </d:Probe>
  </s:Body>
</s:Envelope>

```

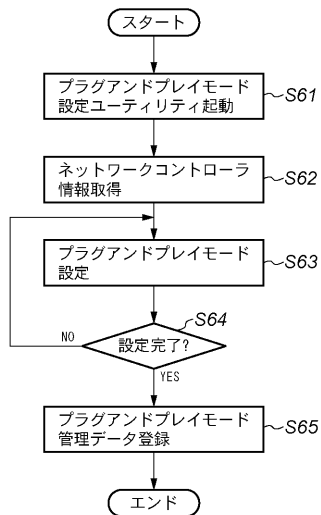
【図 13】

```

<s:Envelope
  xmlns:a="http://schemas.xmlsoap.org/ws/2004/08/addressing"
  xmlns:d="http://schemas.xmlsoap.org/ws/2005/04/discovery"
  xmlns:i="http://printer.example.org/ws/2003/imaging"
  xmlns:s="http://www.w3.org/2003/05/soap-envelope">
  <s:Header>
    <a:Action>
      http://schemas.xmlsoap.org/ws/2005/04/discovery/ProbeMatches
    </a:Action>
    <a:MessageID>
      uuid:e32e6863-ea5e-4ee4-997e-69539d1ff2cc
    </a:MessageID>
    <a:RelatesTo>
      uuid:0a6dc791-2be6-4991-9af1-454778a1917a
    </a:RelatesTo>
    <a:To>
      http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous
    </a:To>
    <d:AppSequence InstanceId="1077004800" MessageNumber="2"/>
  </s:Header>
  <s:Body>
    <d:ProbeMatches>
      <d:ProbeMatch>
        <a:EndpointReference>
          <a:Address>
            uuid:98190dc2-0890-4ef8-ac9a-5940995e6119
          </a:Address>
          </a:EndpointReference>
          <d:Types>i:PrintBasic i:PrintAdvanced</d:Types>
          <d:Scopes>
            ldap:///ou=engineering,o=examplecom,c=us
            ldap:///ou=floor1,ou=b42,ou=anytown,o=examplecom,c=us
            http://itdept/imaging/deployment/2004-12-04
          </d:Scopes>
          <d:XAddr>http://prn-example/PRN42/b42-1668-a</d:XAddr>
          <d:MetadataVersion>75965</d:MetadataVersion>
        </d:ProbeMatch>
      </d:ProbeMatches>
    </s:Body>
  </s:Envelope>

```

【図 14】



【図 15】

◎ ネットワークを選択してください

Wi-Fi ▼

◎ ネットワークデバイスの検索

許可 ▼

禁止

◎ Helloメッセージの応答

禁止 ▼

OK Cancel

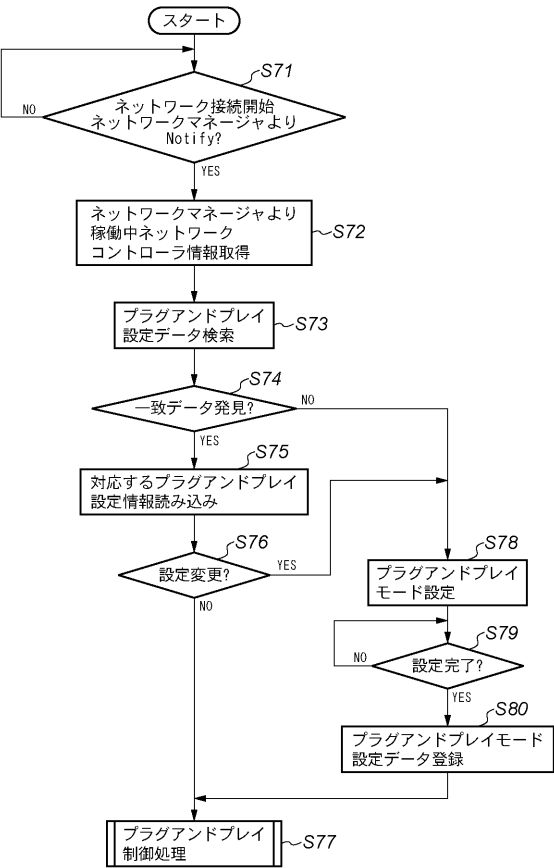
【図 16】

```

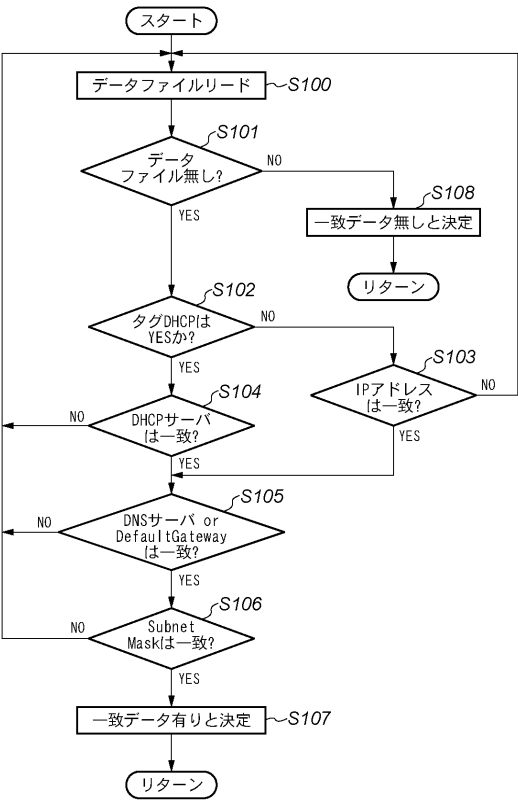
<PnPModeData>
  <NetworkType>Ethernet|Wi-Fi|Bluetooth</NetworkType>
  <PnPSetting>
    <Probr>YES|NO</Probe>
    <Hello>YES|NO</Hello>
  </PnPSetting>
</PnPModeData>

```

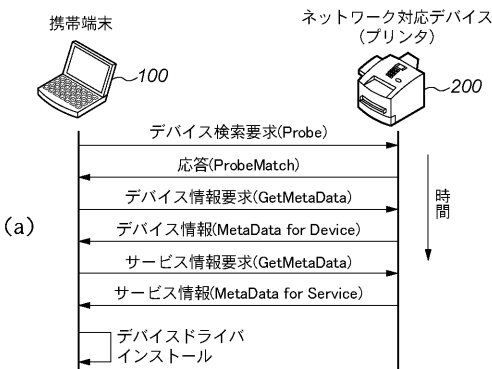
【図 17】



【図 18】

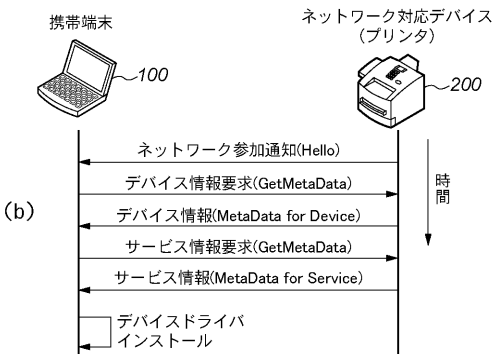


【図 19】



【図 20】

DHCP	Yes	No	Yes
IP Address	....	192.168.3.220	....
SubnetMask	255.255.255.0	255.255.255.0	255.255.0.0
Gateway	192.168.1.150	192.168.3.150	192.168.220.150
DNSServer	192.168.1.200	192.168.3.200	192.168.230.200
DHCPServer	192.168.1.220	.....	192.168.240.220
Probe	Yes	No	No
Hello	Yes	Yes	No



---

フロントページの続き

審査官 千本 潤介

(56)参考文献 特開 2 0 0 3 - 3 1 6 6 5 0 ( J P , A )  
特開 2 0 0 1 - 1 1 7 8 3 4 ( J P , A )  
特開 2 0 0 5 - 1 3 5 4 1 4 ( J P , A )  
特開 2 0 0 3 - 3 2 3 3 6 3 ( J P , A )

(58)調査した分野(Int.Cl. , D B 名)  
G 0 6 F 1 3 / 1 0  
G 0 6 F 1 3 / 0 0  
G 0 6 F 3 / 1 2