

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4087788号
(P4087788)

(45) 発行日 平成20年5月21日(2008.5.21)

(24) 登録日 平成20年2月29日(2008.2.29)

(51) Int.Cl.		F I			
G06F 13/14	(2006.01)		G06F 13/14	330C	
G06F 3/12	(2006.01)		G06F 3/12	C	
G06F 13/10	(2006.01)		G06F 13/10	310E	

請求項の数 5 (全 25 頁)

(21) 出願番号	特願2003-435293 (P2003-435293)	(73) 特許権者	390002761
(22) 出願日	平成15年12月26日(2003.12.26)		キヤノンマーケティングジャパン株式会社
(65) 公開番号	特開2005-196261 (P2005-196261A)		東京都港区港南2丁目16番6号
(43) 公開日	平成17年7月21日(2005.7.21)	(74) 代理人	100076428
審査請求日	平成17年6月28日(2005.6.28)		弁理士 大塚 康徳
		(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	塩山 悠介
			東京都中央区日本橋大伝馬町12番2号
			株式会社ソフトウェアコントロール内

最終頁に続く

(54) 【発明の名称】 情報処理装置及びその制御方法、プログラム

(57) 【特許請求の範囲】

【請求項1】

デバイスを管理するためのデバイスオブジェクトを生成するオペレーティングシステムを備える情報処理装置であって、

前記オペレーティングシステムにより管理されるレジストリ領域から、既に情報処理装置にインストールされているデバイスのベンダー識別情報とプロダクト識別情報を含むデバイス情報を取得するデバイス情報取得手段と、

前記デバイス情報取得手段で取得したデバイス情報に従って、同機種種のデバイスが接続されたプラグアンドプレイ時に、新規にデバイスオブジェクトが前記オペレーティングシステムによって作成されることを禁止するための、前記ベンダー識別情報と前記プロダクト識別情報とを含むプラグアンドプレイ制御情報を生成する生成手段と、

同機種種のデバイスが接続された際に、前記オペレーティングシステムによって同機種種のデバイスのデバイスオブジェクトが新規に作成されることを禁止するために、前記オペレーティングシステムがデバイスオブジェクトを作成する際に参照するレジストリ領域に、前記生成手段で生成されたプラグアンドプレイ制御情報を設定する設定手段とを備えることを特徴とする情報処理装置。

【請求項2】

処理対象となるデバイスを特定するためのデバイス名称を取得するデバイス名称取得手段とを更に備え、

前記ベンダー識別情報とプロダクト識別情報を含むデバイス情報は、前記デバイス名称

取得手段で取得されたデバイス名称に従って取得される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】

既に情報処理装置にインストールされているデバイスの一覧情報を取得するデバイス一覧情報取得手段と、

前記デバイス一覧情報取得手段で取得したデバイスの一覧情報を表示する表示制御手段と、

前記表示制御手段で表示された一覧情報からデバイスを選択する選択手段とを更に備え

、前記デバイス名称は、前記選択手段で選択されたデバイスに従って取得される

ことを特徴とする請求項 2 に記載の情報処理装置。

10

【請求項 4】

デバイスを管理するためのデバイスオブジェクトを生成するオペレーティングシステムを備える情報処理装置の制御方法であって、

前記オペレーティングシステムにより管理されるレジストリ領域から、既に情報処理装置にインストールされているデバイスのベンダー識別情報とプロダクト識別情報を含むデバイス情報を取得するデバイス情報取得工程と、

前記デバイス情報取得工程で取得したデバイス情報に従って、同機種のデバイスが接続されたプラグアンドプレイ時に、新規にデバイスオブジェクトが前記オペレーティングシステムによって作成されることを禁止するための、前記ベンダー識別情報と前記プロダクト識別情報とを含むプラグアンドプレイ制御情報を生成する生成工程と、

20

同機種のデバイスが接続された際に、前記オペレーティングシステムによって同機種のデバイスのデバイスオブジェクトが新規に作成されることを禁止するために、前記オペレーティングシステムがデバイスオブジェクトを作成する際に参照するレジストリ領域に、前記生成工程で生成されたプラグアンドプレイ制御情報を設定する設定工程と

を備えることを特徴とする情報処理装置の制御方法。

【請求項 5】

デバイスを管理するためのデバイスオブジェクトを生成するオペレーティングシステムを備える情報処理装置の制御をコンピュータに実行させるためのプログラムであって、

前記オペレーティングシステムにより管理されるレジストリ領域から、既に情報処理装置にインストールされているデバイスのベンダー識別情報とプロダクト識別情報を含むデバイス情報を取得するデバイス情報取得工程と、

30

前記デバイス情報取得工程で取得したデバイス情報に従って、同機種のデバイスが接続されたプラグアンドプレイ時に、新規にデバイスオブジェクトが前記オペレーティングシステムによって作成されることを禁止するための、前記ベンダー識別情報と前記プロダクト識別情報とを含むプラグアンドプレイ制御情報を生成する生成工程と、

同機種のデバイスが接続された際に、前記オペレーティングシステムによって同機種のデバイスのデバイスオブジェクトが新規に作成されることを禁止するために、前記オペレーティングシステムがデバイスオブジェクトを作成する際に参照するレジストリ領域に、前記生成工程で生成されたプラグアンドプレイ制御情報を設定する設定工程と

40

をコンピュータに実行させることを特徴とするプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、デバイスを管理するための管理情報の設定を制御するデバイス制御情報を作成する情報処理装置及びその制御方法、プログラムに関するものである。

【背景技術】

【0002】

コンピュータのオペレーティングシステム（以下、OS）には、プラグアンドプレイという概念が存在する。これは、コンピュータ本体にハードウェア/外部機器を接続するだ

50

けで、そのハードウェア/外部機器を使用できるようにする仕組みである。例えば、プラグアンドプレイの規格に対応したUSBデバイス(USBインターフェースを有するデバイス(例えば、プリンタ、スキャナ等))は、ハードウェア情報(例えば、機種情報(ベンダーID(Vid)とプロダクトID(Pid))や个体情報(个体ID))をUSBデバイスを特定するための固有識別情報として、コンピュータのOSに伝達し、OSはその固有識別情報を元に、そのUSBデバイスを管理し、使用可能にするために、そのデバイスドライバやデバイス設定情報をインストールしたり、設定したりする。

【0003】

USBデバイスの好適な一例であるUSBプリンタにおいて、米国マイクロソフト社のOSであるWindows(登録商標)では、いままでに接続されたことの無いプリンタが接続された場合、プラグアンドプレイにより取得したハードウェアIDをユーザにわかりやすい名称(以下、プリンタ名称)で表現し、OSが提供するグラフィックユーザインターフェース上にそのプリンタ名称を持ったプリンタオブジェクトを作成する。

10

【0004】

そして、プリンタオブジェクトが既に作成済みであるプリンタが再度接続された場合は、該当するプリンタオブジェクトを有効にする。この場合、ユーザは、プリンタを使用するとき、有効になったプリンタオブジェクトを選択することとなる。

【0005】

ここで、既にプリンタドライバがインストール済みのプリンタと同じ機種情報を有するプリンタ、つまり、同機種の接続したことの無い別個体のUSBプリンタを接続した場合を考える。この場合、OSは个体情報を比較し、既に接続されたことがあるプリンタと同じ機種情報を有するものでも別のUSBプリンタと判断する。その結果、既にインストールされたプリンタオブジェクトの複製を作成して、それを接続されたプリンタに割り当て管理する。

20

【0006】

また、特許文献1では、過去に接続された周辺機器のベンダーID(Vid)とプロダクトID(Pid)とデバイスIDと接続パス情報と、現在接続されている周辺機器の情報とを比較する。そして、比較した情報同士が一致する場合、過去に付与したポート名称と同一のポート名称をその周辺機器に付与する。一方、比較した情報同士が一致しない場合は、新たにポート名称を作成し、現在接続されている周辺機器のベンダーIDと及びデバイスIDと共に接続パス情報と生成したポート名称とをホストコンピュータの記憶装置に記憶することが記載されている。

30

【特許文献1】特開2003-196224

【発明の開示】

【発明が解決しようとする課題】

【0007】

このように、プラグアンドプレイ機能を有するWindows(登録商標)環境では、例えば、同機種のプリンタであっても異なる个体IDを持ったプリンタを新規にコンピュータに接続した場合には、既にインストールされた同機種のプリンタオブジェクトの複製が作成されることになる。

40

【0008】

しかしながら、この方法では同じ機種情報を持つ、つまり、同機種のプリンタを複数所有し、使用しているユーザにとって不便が生じる場合がある。

【0009】

つまり、ユーザにとって同機種のプリンタが複数台あったとしても、それを交互に使う場合、プリンタ設定情報やプリンタオブジェクトは同じものを使用したい要求があるからである。例えば、10名の営業マンが5台のモバイルプリンタを使用する場合、各営業マンは同一のモバイルプリンタを使用するわけではなく、使用したい日に空いている任意のモバイルプリンタを客先等に持参して、自分のノートパソコンに接続して提案資料等を印刷するなどの場合が考えられる。

50

【 0 0 1 0 】

このように、プリンタの個数分プリンタオブジェクトが存在すると、ユーザは、使用するプリンタオブジェクトを適宜切り替えなければならず、また、印刷の設定も個々に設定しなおさなければいけないと言う手間が生じる。つまり、上記の例では、各営業マンは最大5個までのプリンタオブジェクトが自己のノートパソコンに存在してしまい、印刷時に実際に接続されているモバイルプリンタを確認して切り替える手間が生じる。

【 0 0 1 1 】

また、プリンタの故障時にそのプリンタの代替機を使用する場合は、代替機を接続することによって、その故障したプリンタに対するプリンタオブジェクトに加えて、代替機に対応する新しいプリンタオブジェクトが作成されることになる。しかしながら、作成されるプリンタオブジェクトは、故障機と代替機とで区別して表示されるわけではないので、どのプリンタオブジェクトが代替機のものであるかを容易に判別することができない。加えて、同一機種にも関わらず、プリンタ設定等の初期設定を再度行わなければならず手間が生じる。

【 0 0 1 2 】

また、特許文献1では、ベンダーID (V i d) とプロダクトID (P i d) に基づいて、接続される周辺機器を識別しているが、新たな周辺機器が接続された場合には、新たなポートが付与されてしまい、上記例と同様の課題があった。

【 0 0 1 3 】

本発明は上記の課題を解決するためになされたものであり、デバイスを管理するための管理情報を、使用する環境に応じて、好適に管理、設定することができる情報処理装置及びその制御方法、プログラムを提供することを目的とする。

【課題を解決するための手段】

【 0 0 1 4 】

上記の目的を達成するための本発明による情報処理装置は以下の構成を備える。即ち、デバイスを管理するためのデバイスオブジェクトを生成するオペレーティングシステムを備える情報処理装置であって、

前記オペレーティングシステムにより管理されるレジストリ領域から、既に情報処理装置にインストールされているデバイスのベンダー識別情報とプロダクト識別情報を含むデバイス情報を取得するデバイス情報取得手段と、

前記デバイス情報取得手段で取得したデバイス情報に従って、同機種のデバイスが接続されたプラグアンドプレイ時に、新規にデバイスオブジェクトが前記オペレーティングシステムによって作成されることを禁止するための、前記ベンダー識別情報と前記プロダクト識別情報とを含むプラグアンドプレイ制御情報を生成する生成手段と、

同機種のデバイスが接続された際に、前記オペレーティングシステムによって同機種のデバイスのデバイスオブジェクトが新規に作成されることを禁止するために、前記オペレーティングシステムがデバイスオブジェクトを作成する際に参照するレジストリ領域に、前記生成手段で生成されたプラグアンドプレイ制御情報を設定する設定手段と
を備える。

【 0 0 1 5 】

また、好ましくは、処理対象となるデバイスを特定するためのデバイス名称を取得するデバイス名称取得手段とを更に備え、

前記ベンダー識別情報とプロダクト識別情報を含むデバイス情報は、前記デバイス名称取得手段で取得されたデバイス名称に従って取得される。

また、好ましくは、既に情報処理装置にインストールされているデバイスの一覧情報を取得するデバイス一覧情報取得手段と、

前記デバイス一覧情報取得手段で取得したデバイスの一覧情報を表示する表示制御手段と、

前記表示制御手段で表示された一覧情報からデバイスを選択する選択手段とを更に備え

、

10

20

30

40

50

前記デバイス名称は、前記選択手段で選択されたデバイスに従って取得される。

【0023】

上記の目的を達成するための本発明による情報処理装置の制御方法は以下の構成を備える。即ち、

デバイスを管理するためのデバイスオブジェクトを生成するオペレーティングシステムを備える情報処理装置の制御方法であって、

前記オペレーティングシステムにより管理されるレジストリ領域から、既に情報処理装置にインストールされているデバイスのベンダー識別情報とプロダクト識別情報を含むデバイス情報を取得するデバイス情報取得工程と、

前記デバイス情報取得工程で取得したデバイス情報に従って、同機種のデバイスが接続されたプラグアンドプレイ時に、新規にデバイスオブジェクトが前記オペレーティングシステムによって作成されることを禁止するための、前記ベンダー識別情報と前記プロダクト識別情報とを含むプラグアンドプレイ制御情報を生成する生成工程と、

同機種のデバイスが接続された際に、前記オペレーティングシステムによって同機種のデバイスのデバイスオブジェクトが新規に作成されることを禁止するために、前記オペレーティングシステムがデバイスオブジェクトを作成する際に参照するレジストリ領域に、前記生成工程で生成されたプラグアンドプレイ制御情報を設定する設定工程と
を備える。

【0024】

上記の目的を達成するための本発明によるプログラムは以下の構成を備える。即ち、

デバイスを管理するためのデバイスオブジェクトを生成するオペレーティングシステムを備える情報処理装置の制御をコンピュータに実行させるためのプログラムであって、

前記オペレーティングシステムにより管理されるレジストリ領域から、既に情報処理装置にインストールされているデバイスのベンダー識別情報とプロダクト識別情報を含むデバイス情報を取得するデバイス情報取得工程と、

前記デバイス情報取得工程で取得したデバイス情報に従って、同機種のデバイスが接続されたプラグアンドプレイ時に、新規にデバイスオブジェクトが前記オペレーティングシステムによって作成されることを禁止するための、前記ベンダー識別情報と前記プロダクト識別情報とを含むプラグアンドプレイ制御情報を生成する生成工程と、

同機種のデバイスが接続された際に、前記オペレーティングシステムによって同機種のデバイスのデバイスオブジェクトが新規に作成されることを禁止するために、前記オペレーティングシステムがデバイスオブジェクトを作成する際に参照するレジストリ領域に、前記生成工程で生成されたプラグアンドプレイ制御情報を設定する設定工程と
をコンピュータに実行させる。

【発明の効果】

【0025】

本発明によれば、デバイスを管理するための管理情報を、使用する環境に応じて、好適に管理、設定することができる情報処理装置及びその制御方法、プログラムを提供できる。

【発明を実施するための最良の形態】

【0026】

以下、本発明の実施の形態について図面を用いて詳細に説明する。

【0027】

<実施形態1>

図1は本発明の実施形態1のプリンタ制御システムの機能構成を示すブロック図である。

【0028】

このプリンタ制御システムは、ホストコンピュータ100とプリンタA200から構成されている。また、プリンタB400は、プリンタA200と同機種のプリンタである。両者の間はUSBケーブル300により接続され、プリンタA200とプリンタB400

10

20

30

40

50

は交互に切り替えて接続される。

【0029】

尚、実施形態1では、プリンタA200とプリンタB400を交互に切り替えて接続する場合を例に挙げて説明するが、3台以上の同機種のプリンタを適宜切り替えて接続する場合についても、本発明は適用できる。

【0030】

ホストコンピュータ100は、ホストコンピュータ100全体の動作を制御する中央演算処理部101と、記憶部102と、プリンタ制御部103と、インターフェース制御部104と、入力部105と、ディスプレイ表示部106とを備えている。

【0031】

プリンタ制御部103は、OSやユーティリティツール(ソフトウェア)で実現され、ハードディスク、CD-ROM等の記憶媒体である記憶部102に格納されている。

【0032】

ここで、ホストコンピュータ100の構成要素の一例について、図2を用いて説明する。

【0033】

図2は本発明の実施形態1のホストコンピュータの構成要素を示す図である。

【0034】

図2において、ホストコンピュータ100は、ハードウェア構成要素として、例えば、中央演算処理装置(CPU)1101、RAM1102、ROM1103、LANアダプタ1104、ビデオアダプタ1105、マウス1106、キーボード1107、ハードディスク1108、CD-ROMドライブ1109、USBインターフェース1110を有し、これらの各種構成要素は、システムバス1100を介して相互に接続されている。

【0035】

ここで、システムバス1100は、例えば、PCIバス、AGPバス、メモリバス等を意味する。また、図2では、各バス間の接続用チップやキーボードインターフェースや、いわゆるSCSIやATAPIのような入出力用インターフェースを省略している。

【0036】

尚、中央演算処理装置1101は、図1の中央演算処理部101を実現する。また、RAM1102、ROM1103、ハードディスク1108及びCD-ROMドライブ1109は、図1の記憶部102を実現する。また、LANアダプタ1104、ビデオアダプタ1105及びUSBインターフェース1110は、図1のインターフェース制御部104を実現する。また、マウス1106及びキーボード1107は、図1の入力部105を実現する。

【0037】

中央演算処理装置1101は、オペレーティングシステムのプログラムやアプリケーションプログラムに基づいて、四則演算や比較演算等の各種演算、ハードウェアの制御等を行う。RAM1102には、ハードディスク1108やCD-ROMドライブ1109に装着されたCD-ROMやCD-R等の記憶媒体から読み出されたオペレーティングシステムのプログラムやアプリケーションプログラム等が一時的に記憶され、これらは中央演算処理装置1101の制御の元に行われる。

【0038】

ROM1103には、オペレーティングシステムと協働してハードディスク1108等への入出力を司るいわゆるBIOS等が記憶される。LANアダプタ1104は、中央演算処理装置1101によって制御されるオペレーティングシステムの通信プログラムと協働してネットワークを介した外部との通信を行う。

【0039】

ビデオアダプタ1105は、ディスプレイ装置に出力する画像信号を生成する。キーボード1107やマウス1106は情報処理装置への指示を入力するために用いられる。ハードディスク1108は、オペレーティングシステムやアプリケーションプログラム、図

10

20

30

40

50

1のプリンタ制御部103を実現するユーティリティツール等の各種プログラムを記憶している。

【0040】

CD-ROMドライブ1109は、例えば、CD-ROMやCD-RやCD-R/W等の記憶媒体を装着してアプリケーションプログラムをハードディスク1106にインストールするのに用いる。尚、CD-ROMドライブ1109の代わりに、CD-RドライブやCD-R/WドライブやMOドライブ等の各種ディスクドライブを用いても良いことは言うまでもない。

【0041】

図1の説明に戻る。

10

【0042】

プリンタA200は、ファームウェア部201を備えており、機種ID202(Vid、Pid)と個体ID203の2つの識別子を管理している。また、USBインターフェース204を備えている。機種ID202は、プリンタA200の機種を識別している。個体ID203は、プリンタA200の個体を識別する固有の値である。

【0043】

ここで、機種IDの内、Vidは、プリンタのベンダーを識別するためのベンダーIDであり、Pidは、プリンタの機種を識別するためのプロダクトIDである。

【0044】

また、プリンタB400は、プリンタA200と同機種のプリンタで、ファームウェア部401を備え、機種ID402と個体ID403の2つの識別子を管理している。また、USBインターフェース404を備える。プリンタB400はプリンタA200と同ベンダーかつ同機種のプリンタであるため、プリンタA200の機種ID(Vid、Pid)202とプリンタB400の機種ID(Vid、Pid)402は同一である。これに対し、それぞれのプリンタの個体ID203と個体ID403は異なるものである。

20

【0045】

尚、個体ID203及び403には、USBインターフェースを識別するUSB IDを含むようにしてもよい。

【0046】

尚、プリンタA200やプリンタB400の印刷方式は、例えば、インクジェット方式、レーザービーム方式、熱転写方式等の各種印刷方式を利用することができ、利用する印刷方式によって、その印刷方式を実現するためのプリンタエンジンやコントローラ、記録部を有している。

30

【0047】

ここで、プリンタA200やプリンタB400のハードウェア構成について、図3を用いて説明する。

【0048】

図3は本発明の実施形態1のプリンタのハードウェア構成を示す図である。

【0049】

図3において、1900はプリンタの電源部であり、1901は電源のON/OFF操作を実現する電源スイッチである。1902は電源スイッチ1901の操作内容を示す操作信号をMPU1906へ伝達する信号線である。この操作信号は、信号1903への割込信号としてMPU1905へ伝達されるため、MPU1905において、優先的に処理される。1904はプリンタコントローラ1905、MPU1906、ROM1907を接続するデータバスである。

40

【0050】

ROM1907には、MPU1906の動作、処理を記述したプログラムが記憶されている。このプログラムは、一般的にはファームウェアと呼ばれ、図1のプリンタA200のファームウェア部201、プリンタB400のファームウェア部401に相当する。また、ROM1907には、プリンタA200ではその機種ID202及び個体ID203

50

、プリンタB400ではその機種ID402及び個体ID403が記憶されている。

【0051】

1905はプリンタコントローラであり、プリンタ内の各種メモリ（例えば、ROM1907、DRAM1908等）、プリンタ駆動部、インターフェース部等の動作を制御する機能を有する。DRAM1908は、MPU1906が処理を行う場合にデータ等を一時的に記憶する記憶領域を提供するRAMである。

【0052】

1909はプリンタ駆動部内のヘッドモータ1911やフィードモータ1912の制御を行う信号を伝達する信号線である。ヘッドモータ1911は、例えば、インクを吐出することで記録媒体に記録を行うプリントヘッド1913を動作させるモータである。フィードモータ1912は、プリンタにセットされた記録媒体の給紙排紙や、プリントヘッド1913による記録に伴う記録媒体の搬送を行うためのモータである。1910はプリントヘッド1913の動作を制御する制御線である。

【0053】

1914はデータバスであり、インターフェース部内のインターフェースコントローラ1915とプリンタコントローラ1905間のデータの送受信に用いられる。例えば、このデータバス1914によって、ホストコンピュータ100から受信したデータがインターフェースコントローラ1915からプリンタコントローラへ転送される。

【0054】

1918はUSBインターフェース1918であり、プリンタA200ではUSBインターフェース204、プリンタB400ではUSBインターフェース404に相当する。

【0055】

1916はフラッシュメモリであり、USB MPU1919がUSBインターフェース1918を制御するためのプログラムが記憶されている。

【0056】

1917はインターフェースコントローラ1915とフラッシュメモリ1916、USB MPU1919を相互に接続する信号線である。

【0057】

次に、プリンタ制御部103の詳細構成について、図4を用いて説明する。

【0058】

図4は本発明の実施形態1のプリンタ制御部の詳細構成を示す図である。

【0059】

プリンタ制御部103は、オペレーティングシステム1000と、その上で動作するアプリケーション1005、プリンタドライバ1006、ユーティリティツール1007によって構成される。

【0060】

オペレーティングシステム1000は、プラグアンドプレイを実現させるプラグアンドプレイマネージャ1001と印刷要求を管理するスプーラ1002、ユーザインターフェース(UI)1003、デバイスドライバの設定や、アプリケーションの設定等の各種情報を管理しているデータベースであるレジストリ1004を備える。

【0061】

インターフェース制御部1009は、図1のインターフェース制御部104に相当する。また、USBプリンタ1010は、図1のプリンタA200やプリンタB400に対応する。

【0062】

USBプリンタ1010が接続されると、インターフェース制御部1009の一部であるUSBインターフェースは、接続したプリンタの機種IDや個体IDを含むプリンタに関するプリンタ情報を取得して、プラグアンドプレイマネージャ1001に渡す。

【0063】

このプリンタ情報を取得した時に、そのプリンタ情報中から得られる一時的な情報は、

10

20

30

40

50

図2のRAM1102等の記憶部内に保存され、オペレーティングシステム1000等の各種プログラムが使用する。また、ここで取得する一時的な情報は、ここであげた2つの情報以外にも存在し、これについては、図5を用いて後述する。

【0064】

このように、プラグアンドプレイマネージャ1001は、接続されたハードウェアの認識を行う。

【0065】

スプーラ1002は、USBプリンタ1010からのプリントジョブを一時的に記憶し、順次印刷処理を行うための機構である。このスプーラ1002のプロセスは、プリンタのポートごとに行われるものであり、オペレーティングシステム1000が提供する機構である。このスプーラ1002に関係するポート情報は、USBプリンタ1010接続時にプラグアンドプレイマネージャ1001が取得するプリンタ情報の中に含まれ、プラグアンドプレイマネージャ1001はプリンタのポート名とスプーラを関連付ける処理を行う。

10

【0066】

UI1003は、オペレーティングシステム1000がディスプレイに表示するユーザインターフェースである。プラグアンドプレイマネージャ1001は、接続されたUSBプリンタ1010を認識した後、UI1003に対してプリンタオブジェクトの作成処理を行う。

【0067】

ここで、レジストリ1004へ登録されるプリンタ情報の一例について、図5を用いて説明する。

20

【0068】

図5は本発明の実施形態1のプリンタ情報の一例を示す図である。

【0069】

プリンタ情報1201は、複数種類の情報から構成されており、以下、各情報の内容について説明する。

【0070】

まず、1202はプリンタサーバ名である。1203はプリンタ名である。このプリンタ名が、プリンタオブジェクト名となる。1204はネットワーク共有名であり、プリンタがネットワークで共有されているときのプリンタ名を表すものである。1204はポート情報であり、これには、ポート名や番号等が含まれる。

30

【0071】

1205はプリンタドライバ名であり、このプリンタ情報1201に対応するプリンタが使用するプリンタドライバの名称を表すものである。1207はプリンタに対するコメントを表すものである。1208はプリンタ設定情報であり、この詳細については後述する。1209は区切りページファイル名であり、プリンタから印刷する際に印刷ジョブと印刷ジョブの間に注入する区切りのページファイル名を指定するものである。

【0072】

1210はプリントプロセッサ名であり、プリンタが印刷処理を行うときに使用するプリントプロセッサの名称を表すものである。1211はスプールデータ名である。ここで、スプールデータの形式には、例えば、RAWデータやEMFデータ形式等があるが、この名称はどのスプールデータ形式を使用するかを表すものである。

40

【0073】

1212はプリンタパラメータであり、プリンタに与える各種パラメータを表すものである。1213はプリンタのセキュリティ情報であり、プリンタにアクセスが許可されているユーザの情報等を表すものである。1214は印刷ジョブの優先順位情報である。1215は標準ジョブの優先順位情報であり、デフォルトのジョブの優先順位情報を表すものである。1216は印刷可能開始設定時刻、1217は印刷可能終了設定時刻である。1218はプリンタの状態を示す状態情報である。1219はプリンタの保持するジョブ

50

数を表すものである。1220はプリンタの平均印刷時間を表すものである。

【0074】

図4の説明に戻る。

【0075】

プラグアンドプレイマネージャ1001は、接続されたUSBプリンタ1010を認識すると、その旨をオペレーティングシステム1000上で実行されるアプリケーション1005に対して通知を行う。

【0076】

例えば、Windows(登録商標)オペレーティングシステム上では、ウィンドウメッセージと呼ばれる特定のメッセージを起動されているアプリケーション1005や、ユーティリティツール1007に対して発行する。そのメッセージを受けたアプリケーション1005やユーティリティツール1007は個々にプリンタの追加を認識することができる。

【0077】

また、プラグアンドプレイマネージャ1001は、接続されたUSBプリンタ1010の適切なプリンタドライバ1006の読み込みも行う。

【0078】

ホストコンピュータ100にインストールされているユーティリティツール1007では、プラグアンドプレイマネージャ1001によって登録されたUI1003の情報や、レジストリ1004の情報を操作する仕組みを備えている。そして、ユーティリティツール1007は、例えば、Win32APIと呼ばれるオペレーティングシステムが備えているAPI(アプリケーションプログラムインタフェース)関数1008を用いて、上述した情報を操作することが可能である。

【0079】

次に、プリンタA200をホストコンピュータ100に接続した場合に、ホストコンピュータ100が実行するプラグアンドプレイインストールについて説明する。

【0080】

図1において、ホストコンピュータ100にプリンタA200をUSBケーブル300で接続すると、ホストコンピュータ100のインターフェース制御部104は、プリンタA200の機種ID202と個体ID203を含むプリンタ情報を取得する。

【0081】

上述したとおり、このプリンタ情報はインターフェース制御部104を介して、記憶部102(例えば、レジストリ1004やRAM1102等)の一時記憶領域を使用して、プリンタ制御部103に渡される。

【0082】

プリンタ制御部103の一部であるオペレーティングシステム1000が備えるプラグアンドプレイマネージャ1001は、記憶部102より取得したプリンタ情報と、ファームウェア部401(例えば、フラッシュメモリ1916やROM1907等)より取得した機種ID202や個体ID203を基に、接続されたプリンタA200が新規に接続されたプリンタであるか否かを判定する。

【0083】

そして、その判定結果に基づいて、ディスプレイ表示部106で表示するプリンタ管理画面内にプリンタオブジェクトを作成する。また、このプリンタオブジェクトに対応するプリンタのプリンタドライバを記憶部102にインストールする。

【0084】

尚、プリンタオブジェクトとは、少なくとも一台のプリンタ毎に対応する、例えば、プリンタドライバによって用いられる管理情報であり、プリンタアイコンとしても表示可能である。また、この管理情報には、プリンタの種類等を特定することが可能な上記のプリンタ情報が含まれる。

【0085】

10

20

30

40

50

例えば、プリンタオブジェクトは、例えば、Windows（登録商標）においては、レジストリ情報と呼ばれるデータベース（レジストリ1004）の形式で管理されている。そして、記憶部102に記憶されている表示制御手段の一例であるオペレーティングシステム1000が、該レジストリ情報を読み出すことにより、プリンタ管理画面内で生成するプリンタアイコン画像として、プリンタオブジェクトを表示する処理を制御することができる。

【0086】

次に、プリンタA200を切り離し、プリンタB400を接続するとホストコンピュータ100のインターフェース制御部104は、ファームウェア部401よりプリンタB400の機種ID402と個体ID403を含むプリンタ情報を取得する。

10

【0087】

プリンタ制御部103の一部であるオペレーティングシステム1000が備えるプラグアンドプレイマネージャ1001は、記憶部102より取得したプリンタ情報と、ファームウェア部401より取得した機種ID402や個体ID403を元に、接続されたプリンタB400が新規に接続されたプリンタであるか否かを判定する。

【0088】

この場合、プリンタB400の機種ID402は、既にインストールされているプリンタA200の機種ID202と同一であるが、個体IDが異なるため、新規にプリンタオブジェクトを作成して、プリンタ管理画面内に表示する。

【0089】

20

実際には、これらのインターフェース制御部104やプリンタ制御部103の処理は、中央演算処理部101でのソフトウェア実行により実現される。そのプログラムは、OSやユーティリティソフトウェアとして実装するのが一般的であるが、専用ハードウェアで実現されても良い。

【0090】

実施形態1では、Windows（登録商標）環境で実施する例を挙げて説明するが、本発明は、このような構成に限定されるものではない。

【0091】

また、下記のように処理してもよい。ユーティリティツール1007は、デバイスの好適な一例であるプリンタ毎に対応付けられて管理されるデバイス標識情報の一例であるプリンタアイコン（プリンタオブジェクトという場合もある。）を表示部の一例であるディスプレイ表示部106に表示する処理を制御する。

30

【0092】

つまり、ユーティリティツール1007は、プリンタが通信媒体の一例であるUSBインタフェース300を介してホストコンピュータ100に接続されるのに応答して、デバイスのデバイス識別情報の一例であるプリンタ情報を、USBインタフェース300とオペレーティングシステム1000を介して取得する。

【0093】

そして、ユーティリティツール1007は、取得したプリンタ情報をレジストリ1004に登録する処理をオペレーティングシステム1000を介して制御する。そして、オペレーティングシステム1000は、レジストリ1004に登録されたプリンタアイコンをディスプレイ表示部106に表示する。つまり、ユーティリティツール1007は、プリンタアイコンをディスプレイ表示部106に表示する処理を制御する。

40

【0094】

特に、実施形態1では、ユーティリティツール1007は、既にレジストリ1004に登録済みのプリンタ情報と、新たに接続されたプリンタから取得されたプリンタ情報とが一致する場合に、レジストリ1004に新たにプリンタオブジェクトを作成することを禁止する制御情報を設定することが可能である。

【0095】

そして、この設定により、ホストコンピュータ100に接続される各種プリンタの内、

50

任意の同機種プリンタが異なるタイミングで複数回接続されても、対応するプリンタオブジェクトを複数作成することを禁止することが可能である。

【0096】

換言すれば、制御情報が設定されているプリンタのプリンタ情報がプラグアンドプレイ(PnP)インストールによって登録されると、それ以降に同一のプリンタのプラグアンドプレイ(PnP)インストールが発生する状況になっても、そのプリンタのプリンタ情報を作成することが禁止される。つまり、制御情報が設定されているプリンタのプラグアンドプレイに対してオペレーティングシステム1000(プラグアンドプレイマネージャ1001)が実行する処理が制御される。

【0097】

このような構成によって、ユーティリティツール1007は、既存のプリンタ情報と、新たに取得されたプリンタとが一致する場合、既存デバイスオブジェクトを新たに表示する表示処理を禁止する制御を実行することになる。この場合、既存のプリンタオブジェクトとこれに関連付けられた設定が引き継がれ、新たに接続されたプリンタに用いることが可能となる。

【0098】

また、プリンタ情報はプリンタから取得した機種IDや個体ID等のプリンタを識別する識別情報や、取得した識別情報に基づき、オペレーティングシステム1000により生成(登録)された情報により構成されている。

【0099】

さらに、識別情報はオペレーティングシステム1000あるいはユーティリティツール1007からの問い合わせに対してプリンタが識別情報を返すことによって取得しても良いし、プリンタが定期的を送るメッセージに基づきオペレーティングシステム1000が識別情報を取得するようにしても良い。

【0100】

次に、ユーティリティツール1007で、ホストコンピュータ100で利用される各種プリンタの内、任意の同機種プリンタに対して複数のプリンタオブジェクトを作成することを禁止するプリンタオブジェクト制御処理について、図6を用いて説明する。

【0101】

尚、実施形態1のプリンタオブジェクト制御処理は、上述のプリンタ制御部103に備わるユーティリティツール1007によって実行されるが、オペレーティングシステム1000の一機能として制御することも可能である。

【0102】

また、このプリンタオブジェクト制御処理は、ユーティリティツール1007を任意のタイミングで起動することで、ユーザが必要とする時点で、適宜、制御対象のプリンタに対して適用することが可能である。

【0103】

図6は本発明の実施形態1のプリンタオブジェクト制御処理を示すフローチャートである。

【0104】

尚、図6では、例えば、ホストコンピュータ100にプリンタB400を接続した後に、本処理を実行する例について説明するが、本処理は、ホストコンピュータ100にプリンタが接続されていない状態の任意のタイミングで実行することが可能である。

【0105】

まず、ステップS601では、ホストコンピュータ100にインストールされているプリンタの一覧(インストール済プリンタ一覧)情報を取得する。例えば、Windows(登録商標)オペレーティングシステムによると、EnumPrinters()を実行し、ホストコンピュータ100のローカルメモリにインストールされている全プリンタのプリンタ情報を取得する。この際に、ホストコンピュータ100では、全プリンタのプリンタ情報を一覧情報として、例えば、RAM1102のメモリ領域に展開する。

10

20

30

40

50

【0106】

ステップS602では、ステップS601で取得したプリンタ情報からプリンタ名1203(図5)を取得して、プリンタ名一覧画面を表示する。

【0107】

尚、この時点では、ステップS601で取得したプリンタ情報には、同一機種 of プリンタ情報が複数存在している場合があるが、ステップS602で取得するプリンタ名1203は、重複しない異なるプリンタ名(異なるプリンタドライバに対応する)を取得する。あるいは、同一機種 of プリンタ情報が複数存在している場合でも、プリンタ条件等の設定情報が異なっている場合があるため、そのような場合には、同一機種 of プリンタ情報を取得する。

10

【0108】

ここで、プリンタ名一覧画面について、図7を用いて説明する。

【0109】

図7は本発明の実施形態1のプリンタ名一覧画面の一例を示す図である。

【0110】

図7のプリンタ名一覧画面700は、ステップS602で取得したプリンタ名の一覧を示す画面である。図7では、プリンタ名一覧として、「プリンタA~D」と「その他のプリンタ」が表示されている。

【0111】

ここで、「その他のプリンタ」とは、このプリンタ名一覧画面表示時点では、プリンタ名が不明なプリンタであることを示している。つまり、ホストコンピュータ100にプリンタが接続されているものの、そのプリンタに対応するプリンタドライバ1006がインストールされていない場合には、そのプリンタに対応するプリンタ情報が取得できず、正式なプリンタ名をこの時点では取得できないからである。

20

【0112】

また、この「その他のプリンタ」は、プリンタが接続されていない、あるいはプリンタドライバがインストールされていないプリンタを、ユーザが制御対象としたいときに、ユーザが選択できるようにするための表示である。

【0113】

ユーザは、このプリンタ名一覧画面700で表示されるプリンタ名から、プリンタオブジェクトの表示制御(プラグアンドプレイ制御)を適用する制御対象のプリンタを指定することができる。

30

【0114】

特に、図7では、制御対象のプリンタとして、プリンタ名「プリンタB」を指定する状態を示している。そして、制御対象のプリンタを指定している状態で、OKボタン701を入力部105を介して押下すると、その制御対象のプリンタに対する処理として、ステップS603以降の処理を実行する。また、キャンセルボタン702を入力部105を介して押下すると、プリンタ名一覧画面700の表示を終了する。

【0115】

一方、同一機種 of プリンタ情報が複数存在していて、そのプリンタ条件等の設定情報が異なっている場合には、プリンタ名一覧画面700では、それぞれのプリンタ情報に対するプリンタ名を区別可能(例えば、プリンタA(1)、プリンタA(2)、プリンタA(3))に表示する。これにより、複数の同一機種 of プリンタ情報から、所望の設定情報のプリンタ情報のプリンタを制御対象として選択することが可能である。

40

【0116】

図6の説明に戻る。

【0117】

ステップS603では、制御対象のプリンタが選択されたか否かを判定する。制御対象のプリンタが選択された場合(ステップS603でYES)、RAM1102上に選択されたプリンタ名を記憶し、ステップS604へ進む。一方、制御対象のプリンタが選択さ

50

れない場合（ステップS 6 0 3でNO）、制御対象のプリンタが選択されるまで待機する。

【0118】

尚、この判定は、図7のプリンタ名一覧画面700に対する操作に基づいて実行する。つまり、プリンタ名一覧の中で任意のプリンタ名が選択された状態で、OKボタン701が押下された場合に、その選択されているプリンタ名が制御対象のプリンタとして選択されたと判定する。

【0119】

ステップS 6 0 4では、選択された制御対象のプリンタのプリンタドライバが既にインストールされている（つまり、過去に同機種あるいは同一のプリンタが選択されていた）か否かを判定する。プリンタドライバがインストールされている場合（ステップS 6 0 4でYES）、ステップS 6 0 5へ進む。一方、プリンタドライバがインストールされていない場合（ステップS 6 0 4でNO）、例えば、プリンタ名一覧画面700で「その他のプリンタ」が制御対象のプリンタとして選択されている場合、ステップS 6 0 6へ進む。

10

【0120】

ステップS 6 0 5では、選択された制御対象のプリンタのプリンタポートがUSBであるか否かを判定する。

【0121】

尚、USBであるか否かの判定は、制御対象のプリンタに対応するプリンタ情報中のポート情報1205（図5）に基づいて判定する。

20

【0122】

プリンタポートがUSBである場合（ステップS 6 0 5でYES）、ステップS 6 0 8へ進む。一方、プリンタポートがUSBでない場合（ステップS 6 0 5でNO）、処理を終了する。

【0123】

ステップS 6 0 6では、選択された制御対象のプリンタのプリンタドライバのインストールを促す通知を実行する。この通知は、例えば、ダイアログでメッセージを表示させるようにしても、インストールするプリンタドライバをダウンロードするための画面を表示するようにしても良い。

【0124】

ステップS 6 0 7では、プリンタドライバがインストールされたか否かを判定する。プリンタドライバがインストールされた場合（ステップS 6 0 7でYES）、ステップS 6 0 5へ進む。一方、プリンタドライバがインストールされていない場合（ステップS 6 0 7でNO）、プリンタドライバがインストールされるまで待機する。

30

【0125】

ステップS 6 0 8では、プログラム（例えば、オペレーティングシステム1000）中のレジストリ1004に予め記憶されている、プリンタ情報記憶領域を含むUSBデバイス情報記憶領域（例えば、レジストリ1004中のレジストリキー）を取得する。

【0126】

尚、USBデバイス情報記憶領域は、ユーティリティツール1007のプログラム上に予め記述されている情報であっても良い。

40

【0127】

ステップS 6 0 9では、ステップS 6 0 3で選択された制御対象のプリンタのプリンタ名（この場合、プリンタB）とステップS 6 0 8で取得したUSBデバイス情報記憶領域（プリンタ情報記憶領域を含む）に基づいて、ステップS 6 0 3で選択された制御対象のプリンタの機種を示すプリンタ機種情報を取得する。プリンタ機種情報の取得後、そのプリンタ機種情報は、RAM1102のメモリ領域に展開される。

【0128】

例えば、Windows（登録商標）オペレーティングシステムによると、ステップS 6 0 3で選択された制御対象のプリンタ名「プリンタB」は、レジストリキー「HKEY

50

「LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Print\Printers\プリンタB (プリンタ名)\PnPData」に対応する。そして、このレジストリキー中の「DeviceInstanceId」の値から、プリンタBのプリンタ機種情報を特定するための文字列を取得する。
【0129】

具体的には、「DeviceInstanceId」は、例えば、「USBPRINT\プリンタB (プリンタ名)\6&12D78F2A&USB0N (N:整数)」という記述からなっており、この記述の末尾にある「&USB0N」以前を走査して、初めて現れる「¥」までの間の文字列、つまり、「6&12D78F2A」をプリンタ機種情報を特定するための文字列として取得する。

10

【0130】

その後、予め登録されているプリンタ情報記憶領域であるレジストリキー「HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\USB」の子キー内にある「ParentIdPrefix」のデータを適宜取得して、先に取得した文字列「6&12D78F2A」と比較する。

【0131】

比較の結果、文字列が一致する場合、この「ParentIdPrefix」を持つキーの親キーが、プリンタBのプリンタ機種情報 (Vid (ベンダーID) とPid (プロダクトID)) となる。

【0132】

ステップS610では、ステップ609で取得したプリンタ機種情報に基づいて、プラグアンドプレイ (PnP) を制御 (プリンタオブジェクトの表示を制御) するためのPnP制御情報を生成する。Windows (登録商標) オペレーティングシステムでは、名前: IgnoreHWSerNum「Vid」「Pid」 データ: 01 (バイナリ) という形式 (例えば、名前: IgnoreHWSerNum04a91076 データ: 01 (バイナリ)) でレジストリの値を生成することで、本実施形態を実現可能にする。

20

【0133】

尚、PnP制御情報とPnP制御情報のバイナリデータは、RAM1102に展開されても良いし、所定の情報ファイルに格納されるように構成しても良い。

【0134】

また、実施形態1では、特に、PnP制御情報として、制御対象のプリンタに対するポート及びプリンタオブジェクトが、最初 (初回) のプラグアンドプレイで一旦生成された後、次回以降 (初回以降) の同機種あるいは同一のプリンタのプラグアンドプレイ時には、新たにポート及びプリンタオブジェクトがオペレーティングシステム1000によって作成されることを禁止する制御情報を生成する。

30

【0135】

換言すれば、個体IDが異なる同機種のプリンタに対して複数のポート及びプリンタオブジェクトがオペレーティングシステム1000によって作成されることを禁止する制御情報を生成する。

【0136】

このように、実施形態1で生成するPnP制御情報は、制御対象のプリンタのポートの作成を制御する (同機種あるいは同一のプリンタの初回以降のプラグアンドプレイ時には、同一機種に対し複数のポートの作成を禁止する) ポート制御情報と言い換えることができる。また、プリンタオブジェクトの作成も制御する (同機種あるいは同一のプリンタの初回以降のプラグアンドプレイ時には、同一機種に対し複数のプリンタオブジェクトの作成を禁止する) ことになるので、プリンタオブジェクト制御情報と言い換えることもできる。

40

【0137】

ステップS611では、ステップ610で生成したPnP制御情報とPnP制御情報のバイナリデータを、例えば、RAM1102から呼び出し、レジストリ1004に登録す

50

る。例えば、登録する関数には、`RegSetValueEx()`がある。

【0138】

ステップS612では、現在接続されているプリンタ機種情報を取得する。現在接続されているプリンタ機種情報は、例えば、「`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\usbprint\Enum`」に記述されている、現在USB接続を行っているプリンタのプリンタ機種情報を示す文字列を取得することで、取得可能である。

【0139】

実際は、関数`RegQueryValueEx()`を、「`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\usbprint\Enum`」において実行することで、プリンタ機種情報(`Vid`と`Pid`)を取得する。

10

【0140】

ステップS613では、ステップS603で選択された制御対象のプリンタのプリンタ機種情報と、ステップS612で取得したプリンタ機種情報が一致するかどうかを比較する。一致する場合(ステップS613で`YES`)、ステップS616へ進む。一方、一致しない場合(ステップS613で`NO`)、ステップS614へ進む。

【0141】

ステップS614では、選択された制御対象のプリンタのUSB接続を外すこと通知する。この通知は、例えば、ダイアログでメッセージを表示させるようにすれば良い。

20

【0142】

ステップS615では、ステップS614で通知した制御対象のプリンタのUSB接続が外れたか否かを判定する。USB接続が外れた場合(ステップS615で`YES`)、ステップS616へ進む。一方、USB接続が外れない場合(ステップS615で`NO`)、USB接続が外れるまで、待機する。

【0143】

ステップS616では、ステップ603で選択された制御対象のプリンタのプリンタオブジェクトを削除する。また、プリンタオブジェクトを削除する関数としては、例えば、`DeletePrinter()`がある。また、制御対象のプリンタのプリンタオブジェクトが既に複数存在している場合には、それらのプリンタオブジェクトをすべて削除する。

30

【0144】

ここで、ステップS616で、プリンタオブジェクトを削除するのは、ステップS611で`PnP`制御情報を登録すると、ステップ603で選択された制御対象のプリンタが使用できなくなってしまうためである。

【0145】

ステップS617では、ステップ603で選択された制御対象のプリンタのUSBポート情報を削除する。USBポート情報の削除は、例えば、USBポート情報のあるレジストリキーを`SHDeleteKey()`で削除する。また、制御対象のプリンタのUSBポート情報が既に複数存在している場合には、それらのUSBポート情報をすべて削除する。

40

【0146】

ここで、ステップS617で、USBポート情報を削除するのは、USBデバイスの場合、プリンタオブジェクトを削除するだけではUSBポート情報は削除できず、図8のポート名11001のように、ポート情報が残ってしまうので、USBポート情報をレジストリから削除する必要があるからである。

【0147】

これにより、個体IDが異なる同機種の制御対象のプリンタに対しては、複数のポートが作成されることがなくなり、オペレーティングシステム1000に対する処理負荷を軽減することができる。

50

【 0 1 4 8 】

以上説明したように、実施形態 1 によれば、ユーザの指定したプリンタ機種においては、オペレーティングシステム 1 0 0 0 によってプリンタオブジェクトが複数作成されることを禁止できる。そのため、ユーザは、異なる個体 ID の同機種のプリンタを交互に使い分けて使用する場合にも、常に同じプリンタオブジェクトを使用することができるので困惑することもなく、プリンタを接続する毎に新たなプリンタ設定をする必要がなくなる。

【 0 1 4 9 】

また、同機種別固体のプリンタが接続されてもポートが毎回作成されないため、無駄なポートが作成されずオペレーティングシステム 1 0 0 0 の処理の負担を軽減することができる。

10

【 0 1 5 0 】

< 実施形態 2 >

実施形態 1 では、ホストコンピュータ 1 0 0 にインストール済（あるいはインストールを促すことによって設定された）プリンタのプリンタ機種情報に基づいて、制御対象のプリンタに対する P n P 制御情報を生成して登録することで、そのプリンタと同機種あるいは同一のプリンタのプラグアンドプレイ時には、複数のポート及びプリンタオブジェクトがオペレーティングシステム 1 0 0 0 によって作成されることを禁止する構成とした。

【 0 1 5 1 】

これに対し、実施形態 2 では、ホストコンピュータ 1 0 0 にインストールされていないプリンタに対しても、予め情報ファイルにそのプリンタ機種情報が登録されている場合には、そのプリンタに対する P n P 制御情報を生成して登録することで、そのプリンタと同機種あるいは同一のプリンタのプラグアンドプレイ時には、複数のポート及びプリンタオブジェクトがオペレーティングシステム 1 0 0 0 によって作成されることを禁止する構成について説明する。

20

【 0 1 5 2 】

以下、実施形態 2 のプリンタオブジェクト制御処理について、図 9 を用いて説明する。

【 0 1 5 3 】

図 9 は本発明の実施形態 2 のプリンタオブジェクト制御処理を示すフローチャートである。

【 0 1 5 4 】

尚、実施形態 1 の図 6 と同様のステップには、同じ番号を付し、その説明については省略する。

30

【 0 1 5 5 】

まず、ステップ S 6 0 1 ~ ステップ S 6 0 3 の処理によって、制御対象のプリンタが選択される。

【 0 1 5 6 】

次に、ステップ S 7 0 1 では、情報ファイル（図 1 0 ）の有無を判定する。この情報ファイルには、P n P 制御情報を生成するための情報（例えば、図 1 0 に示す、プリンタ機種情報（V i d、P i d）及びそのプリンタドライバ名）が格納されており、例えば、オペレーティングシステム 1 0 0 0 の管理下でユーリティツール 1 0 0 7 で参照可能な所定の初期設定ファイルとして、例えば、R A M 1 1 0 2 に存在する。

40

【 0 1 5 7 】

情報ファイルがある場合（ステップ S 7 0 1 で Y E S）、ステップ S 7 0 2 へ進む。一方、情報ファイルがない場合（ステップ S 7 0 1 で N O）、ステップ S 7 0 5 へ進む。

【 0 1 5 8 】

ステップ S 7 0 2 では、選択された制御対象のプリンタのプリンタドライバ名を取得する。これは、ステップ S 6 0 1 によって、R A M 1 1 0 2 に展開されたプリンタ情報（図 5）から取得する。

【 0 1 5 9 】

ステップ S 7 0 3 では、情報ファイルより、登録されているプリンタ機種情報とプリン

50

タドライバ名を全て取得する。例えば、取得する関数としては、`GetPrivateProfileString()`がある。取得したプリンタ機種情報とプリンタドライバ名は、例えば、RAM1102に展開される。

【0160】

尚、プリンタ機種情報とプリンタドライバ名とは複数機種分記憶されるように構成しても良い。また、情報ファイルには、プリンタ機種情報のバイナリデータを記憶させるように構成しても良い。

【0161】

ステップS704では、選択された制御対象のプリンタのプリンタドライバ名と、情報ファイルのプリンタドライバ名が同じであるか否かを判定する。

10

【0162】

プリンタドライバ名が同じである場合(ステップS704でYES)、ステップS610へ進む。一方、プリンタドライバ名が同じでない場合(ステップS704でNO)、ステップS705へ進む。

【0163】

ステップS704でプリンタドライバ名が同じである場合(ステップS704でYES)、ステップS610及びステップS611を実行する。これにより、制御対象のプリンタに対するPnP制御情報とPnP制御情報のバイナリデータがレジストリ1004に登録される。

【0164】

ステップS705では、選択された制御対象のプリンタのプリンタドライバがインストールされているか否かを判定する。プリンタドライバがインストールされている場合(ステップS705でYES)、ステップS706へ進む。一方、プリンタドライバがインストールされていない場合(ステップS705でNO)、処理を終了する。

20

【0165】

ステップS706では、選択された制御対象のプリンタのプリンタポートがUSBであるか否かを判定する。

【0166】

尚、USBであるか否かの判定は、制御対象のプリンタに対応するプリンタ情報中のポート情報1205(図5)に基づいて判定する。

30

【0167】

プリンタポートがUSBである場合(ステップS706でYES)、ステップS612に進む。一方、プリンタポートがUSBでない場合(ステップS706でNO)、処理を終了する。

【0168】

ステップS706でプリンタポートがUSBである場合(ステップS706でYES)、ステップS612~ステップS617を実行する。これにより、個体IDが異なる同種の制御対象のプリンタに対しては、複数のポートが作成されることがなくなり、オペレーティングシステム1000に対する処理負荷を軽減することができる。

【0169】

一方、ステップS701で情報ファイルが存在しない場合(ステップS701でNO)、あるいはステップS704でプリンタドライバ名が同じでない場合(ステップS704でNO)、ステップS705、ステップS706、ステップ608~ステップ610を実行する。

40

【0170】

ステップS707では、ステップS609で取得したプリンタ機種情報と対応するプリンタドライバ名を情報ファイルに登録する。例えば、登録する関数としては、`WritePrivateProfileString()`がある。

【0171】

尚、情報ファイルが既に作成されている場合には、その情報ファイルにプリンタ機種情

50

報とプリンタドライバ名を登録するように構成してもよく、情報ファイルの内容を全て書き換える（上書き）ように構成しても良い。一方、情報ファイルが作成されていない場合には、新規に情報ファイルを作成して登録してもよい。

【0172】

また、ここでは、情報ファイルに、制御対象のプリンタのプリンタ機種情報とプリンタドライバ名を登録する構成としたが、ユーティリティツール1007が処理の実行時に参照可能な形態であれば、情報ファイルに限定されず、オペレーティングシステム1000管理下の初期設定ファイルや、所定の記憶領域あるいはメモリカード等の外部記憶媒体上に登録することも可能である。

【0173】

更に、情報ファイルは、テキストエディタ等のアプリケーションによって、ユーザが任意のタイミングでその内容を適宜編集加工することができる。これにより、ユーザは、制御対象のプリンタに対するプリンタ情報がホストコンピュータ100に登録されていないくても、将来、制御対象となるプリンタに対するプリンタ情報を事前に登録し、かつそのプリンタに対するPnP制御情報を必要に応じて作成して登録することが可能となる。

【0174】

以上説明したように、実施形態2によれば、情報ファイルに予め制御対象のプリンタに関するプリンタ情報を登録しておくことにより、その情報ファイルに登録されたプリンタ機種においては、プリンタオブジェクトを複数作成することを禁止する設定を実現できるため、ユーザが所望する形態でプリンタを管理することが可能である。

【0175】

また、同機種別固体のプリンタが接続されてもポートが毎回作成されないため、無駄なポートが作成されずオペレーティングシステム1000の処理の負担を軽減することができる。

【0176】

更に、よく使用されているプリンタのプリンタ機種情報を、まとめて事前に情報ファイルに作成して登録することができるので、ユーザのプリンタに係る管理の手間を省き効率の良い設定をおこなうことができる。

【0177】

<実施形態3>

上記実施形態1及び2では、ユーザに制御対象のプリンタを選択させる構成としているが、直前にUSB接続でインストールされたプリンタあるいはよく使われているプリンタを制御対象として選択する構成としても良い。特に、同機種で別固体のプリンタを交互に複数台使用するユーザは、多くの場合、同一設定下でそのプリンタを使用することを意図する可能性が高い。そのため、このような構成にすることで、ユーザは、制御対象のプリンタを選択する操作自体から解放され、より操作効率を向上することができる。

【0178】

ここで、よく使用されているプリンタの判定は、例えば、ホストコンピュータ100にプリンタが接続される毎に生成されるプリンタオブジェクトの数が、所定数（例えば、3台）以上である場合か否かで判定する。

【0179】

尚、よく使用されているプリンタを通常使用するプリンタ（デフォルトプリンタ）としてもよく、通常使用するプリンタを制御対象とすることも可能である。

【0180】

尚、上記実施形態1～3では、ホストコンピュータ100に接続可能なデバイスとして、例えば、プリンタを例に挙げて説明したが、プリンタ以外のデバイス、例えば、スキャナ、デジタルカメラ等の各種デバイスに対しても、本発明を適用することができる。この場合、上述のプリンタオブジェクトやプリンタ情報（プリンタ機種情報）及びPnP制御情報（プリンタ制御情報）は、適用するデバイスに対応するデバイスオブジェクトやデバイス情報（デバイス機種情報）及びPnP制御情報（デバイス制御情報）を用いて、各実

10

20

30

40

50

施形態を実行することになる。

【0181】

以上、実施形態例を詳述したが、本発明は、例えば、システム、装置、方法、プログラムもしくは記録媒体等としての実施態様をとることが可能であり、具体的には、複数の機器から構成されるシステムに適用しても良いし、また、一つの機器からなる装置に適用しても良い。

【0182】

尚、本発明は、前述した実施形態の機能を実現するソフトウェアのプログラム（実施形態では図に示すフローチャートに対応したプログラム）を、システムあるいは装置に直接あるいは遠隔から供給し、そのシステムあるいは装置のコンピュータが該供給されたプログラムコードを読み出して実行することによっても達成される場合を含む。

10

【0183】

従って、本発明の機能処理をコンピュータで実現するために、該コンピュータにインストールされるプログラムコード自体も本発明を実現するものである。つまり、本発明は、本発明の機能処理を実現するためのコンピュータプログラム自体も含まれる。

【0184】

その場合、プログラムの機能を有していれば、オブジェクトコード、インタプリタにより実行されるプログラム、OSに供給するスクリプトデータ等の形態であっても良い。

【0185】

プログラムを供給するための記録媒体としては、例えば、フロッピー（登録商標）ディスク、ハードディスク、光ディスク、光磁気ディスク、MO、CD-ROM、CD-R、CD-RW、磁気テープ、不揮発性のメモ리카ード、ROM、DVD（DVD-ROM、DVD-R）などがある。

20

【0186】

その他、プログラムの供給方法としては、クライアントコンピュータのブラウザを用いてインターネットのホームページに接続し、該ホームページから本発明のコンピュータプログラムそのもの、もしくは圧縮され自動インストール機能を含むファイルをハードディスク等の記録媒体にダウンロードすることによっても供給できる。また、本発明のプログラムを構成するプログラムコードを複数のファイルに分割し、それぞれのファイルを異なるホームページからダウンロードすることによっても実現可能である。つまり、本発明の機能処理をコンピュータで実現するためのプログラムファイルを複数のユーザに対してダウンロードさせるWWWサーバも、本発明に含まれるものである。

30

【0187】

また、本発明のプログラムを暗号化してCD-ROM等の記録媒体に格納してユーザに配布し、所定の条件をクリアしたユーザに対し、インターネットを介してホームページから暗号化を解く鍵情報をダウンロードさせ、その鍵情報を使用することにより暗号化されたプログラムを実行してコンピュータにインストールさせて実現することも可能である。

【0188】

また、コンピュータが、読み出したプログラムを実行することによって、前述した実施形態の機能が実現される他、そのプログラムの指示に基づき、コンピュータ上で稼動しているOSなどが、実際の処理の一部または全部を行ない、その処理によっても前述した実施形態の機能が実現され得る。

40

【0189】

さらに、記録媒体から読み出されたプログラムが、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後、そのプログラムの指示に基づき、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または全部を行ない、その処理によっても前述した実施形態の機能が実現される。

【図面の簡単な説明】

【0190】

50

【図 1】本発明の実施形態 1 のプリンタ制御システムの構成を示すブロック図である。

【図 2】本発明の実施形態 1 のホストコンピュータの構成要素を示す図である。

【図 3】本発明の実施形態 1 のプリンタのハードウェア構成を示す図である。

【図 4】本発明の実施形態 1 のプリンタ制御部の詳細構成を示す図である。

【図 5】本発明の実施形態 1 のプリンタ情報の一例を示す図である。

【図 6】本発明の実施形態 1 のプリンタオブジェクト制御処理を示すフローチャートである。

【図 7】本発明の実施形態 1 のプリンタ名一覧画面の一例を示す図である。

【図 8】本発明の実施形態 1 のポート情報を説明するための図である。

【図 9】本発明の実施形態 2 のプリンタオブジェクト制御処理を示すフローチャートである。 10

【図 10】本発明の実施形態 2 の情報ファイルの構成例を示す図である。

【符号の説明】

【 0 1 9 1 】

1 0 0 ホストコンピュータ

1 0 1 中央演算処理部

1 0 2 記憶部

1 0 3 プリンタ制御部

1 0 4 インターフェース制御部

1 0 5 入力部

20

1 0 6 ディスプレイ表示部

2 0 0 プリンタ A

2 0 1 ファームウェア部

2 0 2 機種 I D

2 0 3 個体 I D

2 0 4 U S B インターフェース

4 0 0 プリンタ B

4 0 1 ファームウェア部

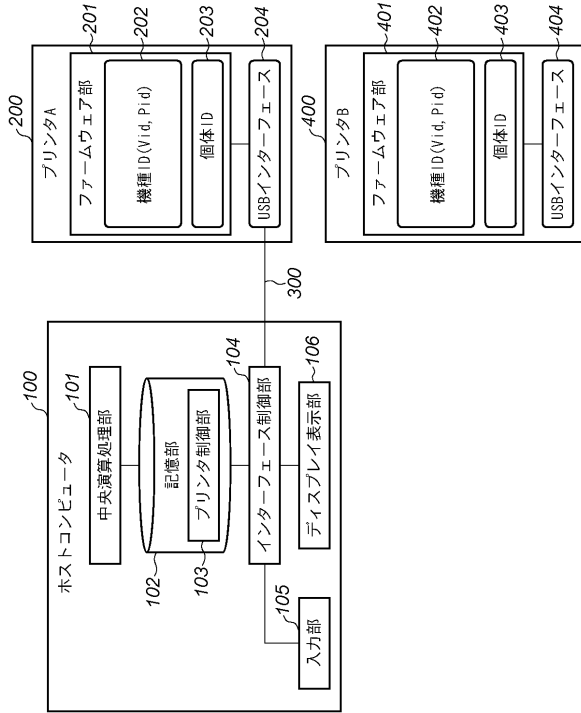
4 0 2 機種 I D

4 0 3 個体 I D

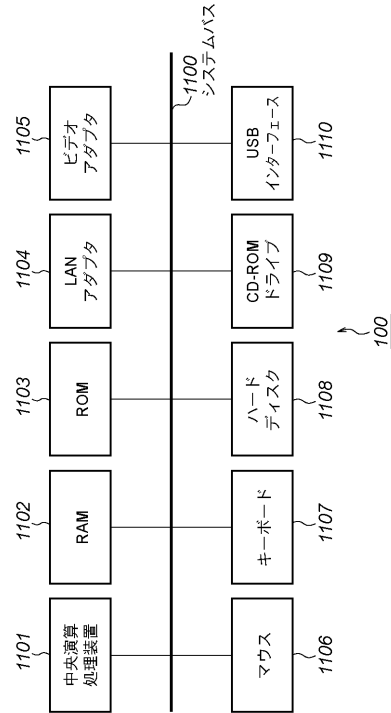
30

4 0 4 U S B インターフェース

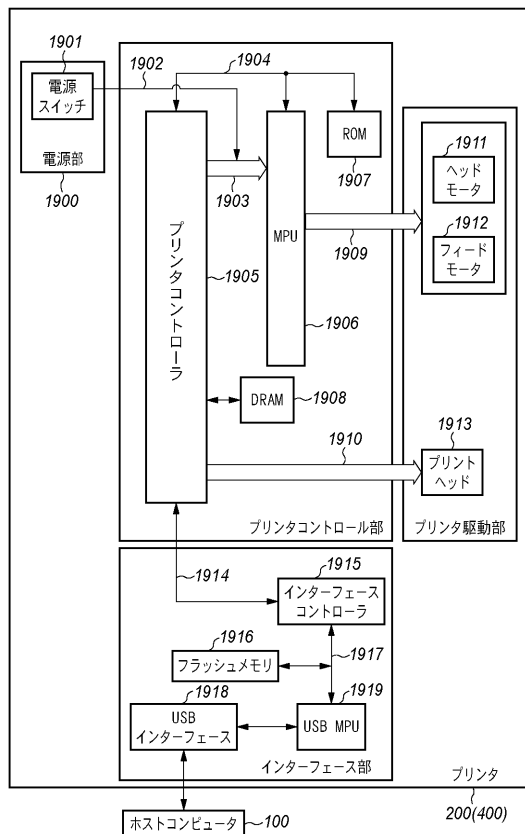
【図1】



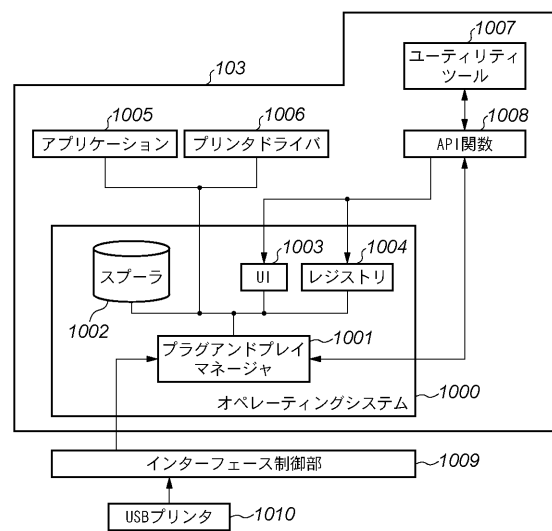
【図2】



【図3】



【図4】

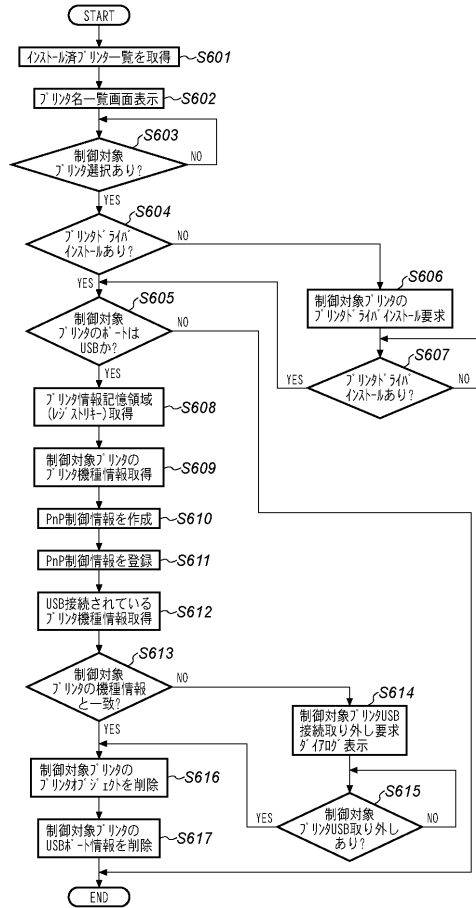


【図5】

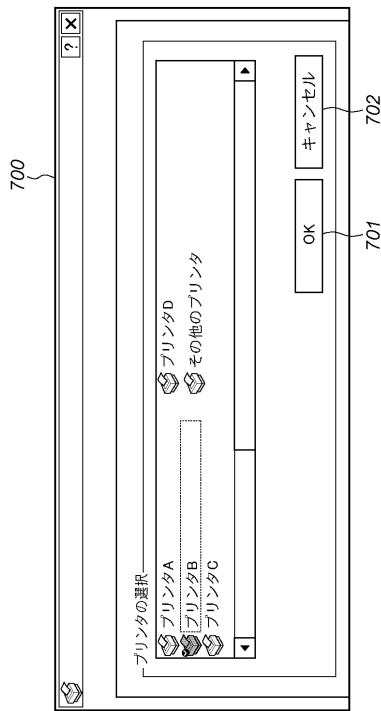
1201

プリンタ情報	
・プリンタサーバ名	1202
・プリンタ名 (プリンタオブジェクト名称)	1203
・ネットワーク共有名	1204
・ポート情報	1205
・プリンタドライバ名	1206
・コメント	1207
・プリンタ設定情報 (DEVMODE)	1208
・区切りページファイル名	1209
・プリントプロセッサ名	1210
・スプールデータ名	1211
・プリンタパラメータ	1212
・プリンタのセキュリティ情報	1213
・印刷ジョブの優先順位情報	1214
・標準ジョブの優先順位情報	1215
・印刷可能開始設定時刻	1216
・印刷可能終了設定時刻	1217
・プリンタの状態情報	1218
・プリンタの保持するジョブ数	1219
・平均印刷時間情報	1220

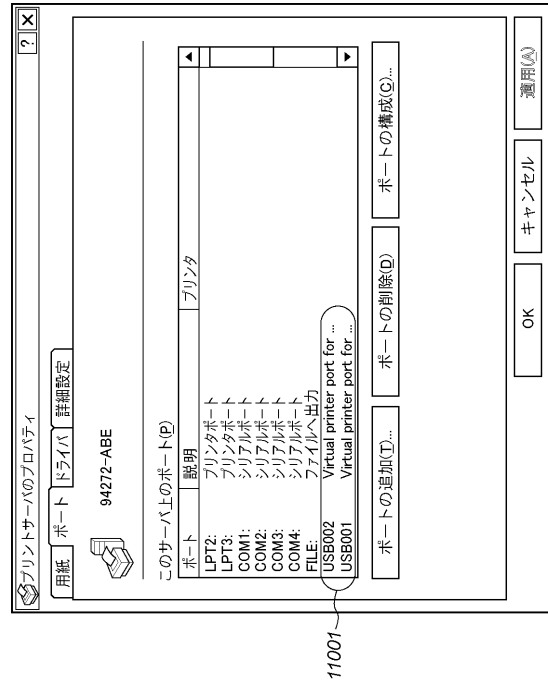
【図6】



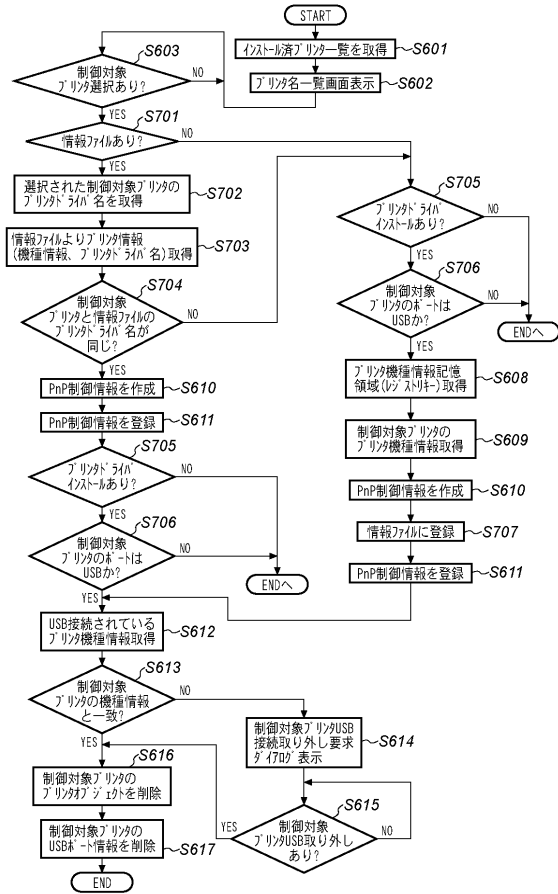
【図7】



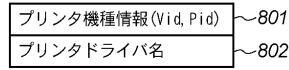
【図8】



【図9】



【図10】



フロントページの続き

(72)発明者 辻 智丈

東京都港区港南2丁目16番16号 キヤノン販売株式会社内

審査官 石井 茂和

(56)参考文献 特開2003-196224(JP,A)
特開2001-312543(JP,A)
特開2003-022225(JP,A)
特開2002-163225(JP,A)
特開2004-252936(JP,A)
特開2003-122353(JP,A)
特開2003-099219(JP,A)
特開2002-123380(JP,A)

(58)調査した分野(Int.Cl., DB名)

G06F 13/14
G06F 3/12
G06F 13/10
WPI(DIALOG)