

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第5080007号

(P5080007)

(45) 発行日 平成24年11月21日(2012.11.21)

(24) 登録日 平成24年9月7日(2012.9.7)

(51) Int.Cl.

F I

G 0 6 F 13/38 (2006.01)

G 0 6 F 13/38 3 5 0

請求項の数 8 (全 20 頁)

(21) 出願番号	特願2006-691 (P2006-691)	(73) 特許権者	500046438
(22) 出願日	平成18年1月5日(2006.1.5)		マイクロソフト コーポレーション
(65) 公開番号	特開2006-195981 (P2006-195981A)		アメリカ合衆国 ワシントン州 9805
(43) 公開日	平成18年7月27日(2006.7.27)		2-6399 レッドモンド ワン マイ
審査請求日	平成20年12月8日(2008.12.8)		クロソフト ウェイ
(31) 優先権主張番号	11/036,893	(74) 代理人	110001243
(32) 優先日	平成17年1月14日(2005.1.14)		特許業務法人 谷・阿部特許事務所
(33) 優先権主張国	米国 (US)	(74) 復代理人	100115624
前置審査			弁理士 濱中 淳宏
		(74) 復代理人	100145388
			弁理士 藤原 弘和

最終頁に続く

(54) 【発明の名称】 アプリケーションサーバ環境におけるUSBデバイス

(57) 【特許請求の範囲】

【請求項1】

リモートクライアントに接続されたクライアントUSBデバイスと対話するアプリケーションプログラムを、前記リモートクライアントの代わりに実行するアプリケーションサーバであって、

前記クライアントUSBデバイスと対話するために前記アプリケーションプログラムが使用する、特定のUSBデバイスのデバイスクラスに固有のインターフェースを有するUSBファンクションドライバと、

前記リモートクライアントに接続された前記クライアントUSBデバイス、および前記特定のクライアントUSBデバイスのデバイスクラスを認識するように構成されているプロキシUSBバスドライバと

を含むサーバドライバスタックを有し、

前記USBファンクションドライバは、前記クライアントUSBデバイスと対話するために前記プロキシUSBバスドライバと通信するように構成され、

前記アプリケーションプログラムは、前記プロキシUSBバスドライバを使用して、前記リモートクライアントに接続された前記特定のクライアントUSBデバイスを区別し、制御すること

を特徴とするアプリケーションサーバ。

【請求項2】

前記プロキシUSBバスドライバは、前記クライアントUSBデバイスと対話するため

10

20

に、前記リモートクライアントにおいて実際のＵＳＢバスドライバと通信するように構成されていることを特徴とする請求項１に記載のアプリケーションサーバ。

【請求項３】

前記サーバドライバスタックは、前記プロキシＵＳＢバスドライバと前記リモートクライアントとの間で通信を提供するように構成されているサーバトランスポートドライバをさらに含むことを特徴とする請求項１に記載のアプリケーションサーバ。

【請求項４】

請求項１に記載のアプリケーションサーバと、
前記リモートクライアントと

を含み、

10

前記リモートクライアントは、前記クライアントＵＳＢデバイスと対話するように構成されているクライアントＵＳＢバスドライバを有すること
を特徴とするクライアントサービスシステム。

【請求項５】

前記プロキシＵＳＢバスドライバは、前記クライアントＵＳＢデバイスと対話するために前記クライアントＵＳＢバスドライバと通信することを特徴とする請求項４に記載のクライアントサービスシステム。

【請求項６】

前記サーバドライバスタックは、前記プロキシＵＳＢバスドライバと前記リモートクライアントとの間で通信を提供するように構成されているサーバトランスポートドライバをさらに含むことを特徴とする請求項４に記載のクライアントサービスシステム。

20

【請求項７】

前記リモートクライアントはクライアントドライバスタックを有し、前記クライアントドライバスタックは、

前記クライアントＵＳＢデバイスと対話するように構成されている前記クライアントＵＳＢバスドライバと、

前記クライアントＵＳＢバスドライバと前記アプリケーションサーバとの間で通信を提供するように構成されているクライアントトランスポートドライバと

を含むことを特徴とする請求項６に記載のクライアントサービスシステム。

30

【請求項８】

前記クライアントサービスシステムは、クライアント／サーバリモートサービスシステムであり、

前記リモートクライアントは、

前記クライアントＵＳＢデバイスと対話するように構成されている前記クライアントＵＳＢバスドライバと、

前記クライアントＵＳＢバスドライバが前記クライアントＵＳＢデバイスと対話する際に経由するＵＳＢホストコントローラドライバと

を含むクライアントドライバスタックを有すること

を特徴とする請求項４に記載のクライアントサービスシステム。

40

【発明の詳細な説明】

【技術分野】

【０００１】

本発明は一般に、ユニバーサルシリアルバス（ＵＳＢ）デバイスと通信するコンピュータ、システム、および方法に関し、より具体的には、ユニバーサルシリアルバス（ＵＳＢ）デバイスのアクセスおよび制御を、サーバコンピュータなどのコンピュータへ導くことに関する。

【背景技術】

【０００２】

本件特許出願人によって提供されているTerminal Serviceなどの、リモートクライアントアクセス用のプラットフォームおよびシステムによって、コンピュー

50

タは、アプリケーションサーバによってホストされ、アプリケーションサーバに常駐しているアプリケーションプログラムに、遠隔地からアクセスすることができる。リモートクライアントアクセスシステムでは、クライアントコンピュータは通常、サーバコンピュータに依存して、そのサーバコンピュータに常駐しているアプリケーションプログラムを介してコンピューティング機能を提供する。アプリケーションプログラムの例としては、ワードプロセッシングプログラム、マルチメディアプログラム、およびデータ管理プログラムなどがある。

【 0 0 0 3 】

アプリケーションサーバコンピュータは、ホストコンピュータまたはターミナルサーバと呼ばれることがある。クライアントコンピュータは、リモート端末、リモートクライアント、またはシンクライアントと呼ばれることがある。クライアントコンピュータは、主にユーザインターフェース、言い換えると、ユーザおよびデバイス／Ｏとの対話用に使用される。クライアントコンピュータ上のソフトウェアは、通常は一般的なものであり、あるいはアプリケーション特有のものではなく、一般にオペレーティングシステムと、リモートクライアントアクセス環境をサポートするソフトウェアを含む汎用ソフトウェアとから構成される。サーバコンピュータにおけるソフトウェアは通常、データベースへのアクセス、文書処理、製図、および他の多くのタイプの用途などの、特定の機能を提供する専用のアプリケーションソフトウェアを含む。クライアントコンピュータとアプリケーションサーバの間で通信されるデータの大部分は、グラフィックスデータ、キーストローク、マウスの動きなどの、ユーザインターフェースに関連するコマンドおよびデータ、ならびにクライアントコンピュータに配置されているハードウェアデバイスに関連するコマンドおよびデータを含む。

【 0 0 0 4 】

アプリケーションサーバおよびクライアントは通常、本件特許出願人によって規定されている R D P (remote desktop protocol) などの所定の通信プロトコルを使用して、情報を相互に通信またはやり取りする。T C P / I P などのより低位のネットワークプロトコルも含まれる。

【 0 0 0 5 】

リモートクライアントアクセスシステムの利点は、大半の機能および計算がサーバコンピュータにおいて行われるため、クライアントコンピュータのパワーを相対的に低く抑えることができることである。アプリケーションサーバは、典型的なデスクトップコンピュータよりも高価な場合が多いが、１台のアプリケーションサーバコンピュータは、より安価な多くのクライアントにサービスを提供することができる。

【 0 0 0 6 】

いくつかのシステムにおける別の利点として、データは、サーバコンピュータの物理ロケーションに常駐することができ、アプリケーションプログラムは、そのロケーションにおいてこれらのデータに基づいて動作することができ、これらのデータを相対的に低速な通信リンクを介してクライアントコンピュータへ転送する必要はなく、クライアントコンピュータの物理ロケーションではユーザインターフェースが実装されるだけである。

【 0 0 0 7 】

クライアントコンピュータは、U S B (ユニバーサルシリアルバス)ポートを有する場合が多く、このポートには周辺デバイスが接続され、このようなデバイスは、U S B キーボードの場合のようにユーザインターフェースに関連する場合が多い。多くの場合、サーバコンピュータにおいて実行されるアプリケーションは、このようなU S B デバイスにアクセスして対話する必要がある。U S B デバイスの他の例としては、デジタルカメラ、ドキュメントスキャナ、外部ディスクドライブ、およびメディアリーダーなどがある。アプリケーションサーバによってホストされるさまざまなアプリケーションは、これらのハードウェアデバイスと対話する必要がある場合がある。

【 0 0 0 8 】

典型的な W i n d o w s (登録商標) ベースのデスクトップ環境では、ローカルアプリ

10

20

30

40

50

ケーションは、ドライバスタックと呼ばれる一連のドライバを介してUSBデバイスと通信する。体系化および再利用上の目的から、スタックのコンポーネント間でさまざまな役割が分担されている。たとえばいくつかのドライバは、一般的なクラスのUSBデバイスのすべてまたは一部に対して汎用性があり、多くの異なるデバイスと通信するために使用または再利用することができる。他のドライバは、特定のデバイスに対する固有の機能を実装し、特定のハードウェア専用設計される場合が多い。

【0009】

図1は、アプリケーションサーバのクライアントとしてではなく、スタンドアロンのコンピュータとして機能するコンピュータ内に実装される典型的なUSBアーキテクチャを示している。図1は、ハードウェアコンポーネントとソフトウェアコンポーネントの間の論理的な通信を示している。

10

【0010】

図1のシステムは、コンピュータ100およびUSBデバイス105を含む。USBデバイス105は、物理的なUSBポート（図示せず）を介してコンピュータ100に接続されている。アプリケーションプログラム110は、コンピュータ上で実行され、ドライバスタック115を介してUSBデバイス105と対話する。この例のドライバスタック115は、3つのUSBドライバを有する。

【0011】

ドライバスタック115の最も下のレベルでは、USBホストコントローラ120が、コンピュータ内のUSBハードウェア（図示せず）と直接通信し、またそのハードウェアを介してUSBデバイス105と通信する。その上では、（ハブまたはハブドライバとも呼ばれる）低位のUSBバスドライバ125が、USBホストコントローラ120と通信し、USBデバイスのパワー、列挙（enumeration）、およびさまざまなUSBトランザクションを管理する。これらのドライバは、双方ともWindows（登録商標）オペレーティングシステムの一部であり、すべてのUSBデバイスに対して汎用性を有しており、これらのドライバは、コンピュータ100に接続されるUSBデバイスのタイプに応じて交換または変更する必要はない。

20

【0012】

ドライバスタック115はまた、USBファンクションデバイスドライバ130を含む。USBファンクションデバイスドライバ130は、特定のデバイスまたはデバイスのクラス用にカスタマイズされる。結果として、使用されている実際のUSBデバイスに応じて、さまざまなファンクションドライバがロードされる。USBファンクションドライバは、USBデバイスファンクションドライバ、クラスドライバ、あるいはカスタムドライバとも呼ばれる。

30

【0013】

図1は、単一のアプリケーションプログラム110しか示していないが、コンピュータ100は、通常は複数のアプリケーションプログラムを有し、そのうちの任意の1つまたは複数のアプリケーションプログラムは、図示されている単一のドライバスタック115を介してUSBデバイス105と対話するように構成することができる。アプリケーションプログラム110は、ワードプロセッシングプログラム、ゲームプログラム、あるいは他のさまざまなタイプの任意のプログラムとすることができる。

40

【0014】

図2は同様の例を示しており、コンピュータ200は、複数のアプリケーションプログラム205および210と、複数のUSBデバイス215、220、および225とを有する。この例では、アプリケーションプログラム205は、USBデバイス215と対話し、その一方でアプリケーションプログラム210は、USBデバイス220および225と対話する。

【0015】

前の例と同様に、通信は、USBドライバスタック230によって実施される。USBドライバスタック230は、USBデバイス215、220、および225のそれぞれと

50

通信するUSBホストコントローラ235およびUSBバスドライバ240を含む。USBドライバスタック230は、USBデバイス215、220、および225のそれぞれに対応する複数のUSBファンクションデバイスドライバ245、250、および255をさらに含む。既に説明したように、これらのUSBファンクションデバイスドライバ245、250、および255のそれぞれは、特定のタイプまたはクラスのUSBデバイスに基づいて選択され、ロードされる。

【0016】

図示されているように、1つのアプリケーションプログラムは、複数のUSBデバイスと対話することができる。複数のUSBデバイスが組み込まれている場合、単一のコンピュータが複数のUSBドライバスタックを有することも可能だが、それらのUSBデバイスは、一般に同一のホストコントローラドライバおよびバスドライバを使用する。異なる種類のUSBポートが使用されている場合、別々のポートドライバと「ミニポート」ドライバをロードすることができるが、一般にUSBドライバスタック全体は、同じまま変わらない。

10

【0017】

図3は、従来技術のリモートクライアント/サーバアーキテクチャにおけるUSBデバイスの使用を示している。この例では、クライアントコンピュータ300は、そのUSBポートに1つまたは複数のUSBデバイス305(1)~305(N)が接続されている。クライアントコンピュータ300はまた、オペレーティングシステム、またはそのようなUSBデバイス用の他の低位のサポートを有する。詳細には、リモートクライアントコンピュータ300は、USBホストコントローラ310を含み、このUSBホストコントローラ310は、USBデバイス305と接続している。

20

【0018】

クライアントコンピュータ300は、自分自身とサーバコンピュータ315の間で転送されるコマンドおよびデータにตอบสนองして、ユーザの対話を実行する。1つまたは複数のアプリケーションプログラム320が、サーバコンピュータ315に常駐し、その上で実行される。遠隔地に配置されているUSBデバイス305をサポートするために、サーバコンピュータ315のオペレーティングシステムまたは他のサポートソフトウェアは、USBサポートを含む。詳細には、サーバコンピュータ315は、USBファンクションデバイスドライバ325(1)~325(N)、USBバスドライバ330、およびUSBホストコントローラ335を含む。

30

【0019】

USBファンクションデバイスドライバ325は、前述のUSBファンクションデバイスドライバ130と同様の機能を実行する。USBファンクションデバイスドライバ325は、USBバスドライバ330と通信する。同じように、USBバスドライバ330は、前述のUSBバスドライバ125と同様の機能を実行する。サーバコンピュータ315は、USBホストコントローラ335をさらに含み、このUSBホストコントローラ335は、クライアントコンピュータ300のUSBホストコントローラ310に接続されているか、またはそのUSBホストコントローラ310と通信する。USBホストコントローラ310とUSBホストコントローラ335の間の通信は、直接の物理的な接続を介して行われるか、またはこの2つを接続する中間ネットワーク(intermediate network)を含むことができる。USBホストコントローラ335は、USBホストコントローラ310との通信を介してUSBデバイス305と通信し、またUSBデバイス305にアクセスする。この例では、クライアントコンピュータ300は、サーバコンピュータ315に完全に依存して、USBデバイス305用のアプリケーションプログラム320を介して、必要な機能をすべて提供する。

40

【発明の開示】

【発明が解決しようとする課題】

【0020】

図1に示されているスタンドアロンのコンピュータの動作とは対照的に、図3のクライ

50

アントサーバの構成では、アプリケーションプログラム320の形態の機能は、すべてサーバコンピュータ315において提供される。すなわち、クライアントコンピュータ300は、USBデバイス305用のすべての機能に関してサーバコンピュータ315に依存している。この構成では、クライアントコンピュータ300は、USBデバイス305のいずれに対しても何の機能も提供しないか、あるいは提供することができない。

【課題を解決するための手段】

【0021】

1つまたは複数のユニバーサルシリアルバス(USB)デバイスが、クライアントコンピュータにおいて接続される。特定のUSBデバイスに対するアクセスおよび制御が、サーバコンピュータに提供され、特定のUSBデバイスからサーバコンピュータへの個別の通信パスが、クライアントコンピュータにおいて提供される。サーバコンピュータからのコマンドはUSBデバイスにおいて受信され、データは特定のUSBデバイスからサーバコンピュータへ送信される。

10

【0022】

サーバコンピュータは、クライアントコンピュータに接続されている特定のUSBデバイスを認識し、特定のUSBデバイス用の個別の通信パスが、サーバコンピュータにおいて提供され、サーバコンピュータは、USBデバイスにコマンドを送信し、サーバコンピュータにおける通信パスを介してUSBデバイスからデータを受信する。

【0023】

添付の図を参照して、詳細な説明を行う。図では、参照番号の最も左側の(1つまたは複数の)数字によって、その参照番号が最初に登場した図を識別する。別々の図において同じ参照番号を使用して、同様のまたは同一のアイテムを表している。

20

【発明を実施するための最良の形態】

【0024】

以降の開示は、特定のUSBデバイスに対するアクセスおよび制御をアプリケーションサーバコンピュータへ導くまたは導き直すプロセスについて説明する。USBデバイスは、アプリケーションサーバコンピュータと通信するクライアントコンピュータにおいて接続される。

【0025】

(クライアント-サーバコンピュータシステム)

30

図4は、リモートクライアントアクセスシステム400を示している。システム400は、アプリケーションサーバコンピュータ405および1つまたは複数のクライアントコンピュータを含む。クライアントコンピュータにおいて接続されているユニバーサルシリアルバス(USB)デバイスに対するアクセスおよび制御は、アプリケーションサーバコンピュータ405へ導くか、または導き直すことができる。詳細には、クライアントコンピュータにおけるUSBデバイスは、アプリケーションサーバコンピュータ405によって選択的にアクセスおよび制御される。システム400は、本件特許出願人によって提供または規定されているTerminal Serviceシステムとすることができ、複数のクライアントコンピュータは、1つまたは複数のUSBデバイスに関する機能、とりわけアクセスおよび制御を提供するすべてまたは一部のアプリケーションプログラムに関して、アプリケーションサーバコンピュータ405に依存する。この例では、クライアントコンピュータ410が示されており、これは他のクライアントコンピュータの例である。

40

【0026】

ネットワーク415は、アプリケーションサーバコンピュータ405とクライアントコンピュータ410を接続する。ネットワーク415は、有線ベースのテクノロジーと無線テクノロジーの双方を含むこのようなネットワーキングコンテキストをサポートする複数の方法で実装することができる。本発明の態様は、1つの特定のネットワークアーキテクチャまたはネットワークテクノロジーに限定されるものではない。システム400は、モデムを介した直接のダイヤルアップ、企業LAN(ローカルエリアネットワーク)、WA

50

N (ワイドエリアネットワーク)、およびインターネットを含むさまざまなアーキテクチャを代表するものである。ネットワーク 415 は、サーバコンピュータ 405 を 1 つまたは複数のクライアントコンピュータ (たとえば、クライアントコンピュータ 410) に接続する。さらにサーバコンピュータ 405 とクライアントコンピュータの間におけるネットワーク 415 の接続は、TCP/IP (transmission control protocol over Internet protocol) などのトランスポートプロトコルを実装することができる。

【0027】

サーバコンピュータ 405 は、本件特許出願人によって提供されている Windows (登録商標) Server 2003 オペレーティングシステムなどのオペレーティングシステムと共に実装される。サーバコンピュータ 405 およびクライアントコンピュータ 410 は、相互にデータまたは情報をやり取りする (すなわち通信する) ために、本件特許出願人によって規定されている RDP (remote data protocol) などの通信プロトコルを実装することができる。このような通信プロトコル、とりわけ RDP の使用は、Terminal Services システムなどによって、リモートクライアントアクセスシステムのコンテキストにおいて実施することができる。

10

【0028】

クライアントコンピュータ 410 は、汎用 PC (パーソナルコンピュータ)、ラップトップ PC、タブレット PC などとすることができ、本件特許出願人による Windows (登録商標) ブランドのオペレーティングシステムなどのオペレーティングシステムを実装することができる。クライアントコンピュータ 410 はスタンドアロンのコンピュータであり、主にインターフェースによってアプリケーションサーバコンピュータ 405 に接続して、クライアントコンピュータ 410 にローカルに保存されていないファイルまたは他の情報 (たとえば、アプリケーションサーバコンピュータ 405 に常駐しているアプリケーションプログラム) にアクセスする。

20

【0029】

クライアントコンピュータ 410 は、USB 1.0、1.1、および 2.0 などの既存の従来の、および将来の USB 標準をサポートする 1 つまたは複数の USB ポートを備えている。これらの USB 標準によって、ポート (コンピュータインターフェース) を分割し、USB デバイスを「デジーチェーン式」に接続することができる。すなわち、クライアントコンピュータ 410 上の 1 つのポートを複数のポートへと「分割」することができ、これによって複数のデバイスが同一のポートを使用することができる。USB ポートは、クライアントコンピュータ 410 内の特定の USB ハブに接続することができる (すなわち、USB 1.1 デバイスは USB 1.1 ハブに接続され、USB 2.0 デバイスは USB 2.0 ハブに接続される)。

30

【0030】

USB ハブは、USB 1.1 ホストコントローラや USB 2.0 ホストコントローラなどのホストコントローラに接続される。場合によっては、ホストコントローラは、複数の USB 標準をサポートすることができる (たとえば 1 つの USB ホストコントローラが、USB 1.1 および 2.0 をサポートする)。ホストコントローラは、クライアントコンピュータ 410 の PCI (peripheral component interconnect) バスなどのバスに接続される。PCI バスは、クライアントコンピュータ 410 の中央処理装置 (CPU) すなわちプロセッサによってアクセスされる。クライアントコンピュータ 410 のハードウェアアーキテクチャについては、以降で図 8 を参照してさらに論じる。

40

【0031】

クライアントコンピュータ 410 は、1 つまたは複数の USB デバイス 420 (1)、420 (2)、および 420 (N) に接続される。USB デバイス 420 は、デジタルカメラ、ビデオカメラ、ハードディスク記憶装置、デジタルメディアレコーダ、プリンタ、およびスキャナを含むが、これらには限定されない。USB デバイス 420 などの USB デバイスは、特定のデバイスクラスによって定義される。すなわちデジタルカメラ、ビデオカメラ、ハードディスク記憶装置、デジタルメディアレコーダ、プリンタ、スキャナな

50

どに対して、個別のデバイスクラスがある。クライアントコンピュータ 410 および USB デバイス 420 は、それぞれ USB 接続 425 (1)、425 (2)、および 425 (N) によって個別に接続される。USB 接続 425 は、前述のようにクライアントコンピュータ 410 において USB ポートの 1 つまたは複数に接続される。

【0032】

(クライアントおよびサーバのコンピュータアーキテクチャ)

図 5 は、USB デバイス 420 のための機能がアプリケーションサーバコンピュータ 405 において提供される、アプリケーションサーバコンピュータ 405 およびクライアントコンピュータ 410 における最上位のアーキテクチャの例を示している。

【0033】

クライアントコンピュータ 410 およびアプリケーションサーバコンピュータ 405 は、そのそれぞれのクライアントトランスポートドライバ 500 およびサーバトランスポートドライバ 505 を介して相互に通信する。図 1 において前述したように、ネットワーク 415 などのネットワークは、サーバコンピュータ 405 とクライアントコンピュータ 410 を接続することができる。さらにクライアントコンピュータ 410、サーバコンピュータ 405、およびネットワーク (たとえばネットワーク 415) は、システム 400 などのより大規模なリモートクライアントアクセスシステムの一部とすることができる。

【0034】

この例では、クライアントコンピュータ 410 は、アプリケーションサーバコンピュータ 405 に依存して、USB デバイス 420 のための機能を提供する。アプリケーションサーバコンピュータ 405 に常駐するアプリケーションプログラム 510 は、1 つまたは複数の USB デバイス 420 のための機能を個別に提供する。1 つまたは複数の USB デバイス 420 のためのこのような機能は、さまざまなリモートクライアントアクセスシステムのシナリオをサポートして、アプリケーションサーバコンピュータ 405 によって提供することができる。

【0035】

1 つの典型的なシナリオは、クライアントコンピュータ 410 が、USB デバイスをサポートするための機能 (すなわち、アプリケーションプログラム) をまったく有していない場合である。このシナリオは、クライアントコンピュータ 410 がシンクライアントであり、サーバコンピュータ 405 によって提供されるすべての機能に依存している場合が典型である。

【0036】

別のシナリオは、クライアントコンピュータ 410 が、USB デバイスをサポートする上で限られた機能または選択された数のアプリケーションプログラムしか有していない場合である。しかし、クライアントコンピュータ 410 に常駐している限られた数のアプリケーションプログラムは、1 つまたは複数の USB デバイス 420 を個別にサポートするアプリケーションプログラムを含まない。サーバコンピュータ 405 は、1 つまたは複数の USB デバイス 420 によって必要とされる機能を提供する特定のアプリケーションプログラム 510 を含む。1 つまたは複数の USB デバイス 420 の制御は、クライアントコンピュータ 410 のアーキテクチャの PNP (plug and play) 機能を介して決定することができる。具体的には、USB バスドライバ 515 は、1 つまたは複数の USB デバイス 420 を認識し、その 1 つまたは複数の USB デバイス 420 の存在を他のコンポーネント (たとえば、クライアントコンピュータの CPU)、ドライバ、またはソフトウェアモジュールに伝える。USB ホストコントローラ 520 は、USB バスドライバとインターフェースで接続され、USB デバイス 420 への物理的なまたは論理的な接続を提供する。他のコンポーネント、ドライバ、またはソフトウェアモジュールによって、その 1 つまたは複数の USB デバイス 420 をサポートするための許容可能なアプリケーションプログラムがクライアントコンピュータ 410 に常駐していないと判断された場合、その 1 つまたは複数の USB デバイス 420 に対する制御は、アプリケーションサーバコンピュータ 405 へ導かれる。

10

20

30

40

50

【 0 0 3 7 】

他の典型的なシナリオでは、クライアントコンピュータ 4 1 0 は、1 つまたは複数の U S B デバイス 4 2 0 をサポートする常駐のアプリケーションプログラムを含むことができるが、複数の理由の 1 つから、クライアントコンピュータ 4 1 0 におけるユーザは、1 つまたは複数の U S B デバイス 4 2 0 のサポートおよび機能をアプリケーションサーバコンピュータ 4 0 5 へ導くことを決定することができる。このユーザによって開始されるオペレーションは、1 つまたは複数の U S B デバイス 4 2 0 に対するアクセスおよび制御をアプリケーションサーバコンピュータ 4 0 5 に提供する。

【 0 0 3 8 】

アプリケーションプログラム 5 1 0 は、U S B ファンクションデバイスドライバ 5 2 5 と通信する。U S B ファンクションデバイスドライバ 5 2 5 は、特定の U S B デバイスのデバイスクラスに固有のものである。U S B ファンクションデバイスドライバ 5 2 5 は、U S B デバイスのクラスを表す特定の F D O (functional device object) を作成する。この特定の F D O は、U S B デバイスのクラスを表すデータオブジェクトであり、アプリケーションプログラム 5 1 0 へのインターフェースを公開または提供する。U S B ファンクションデバイスドライバ 5 2 5 上に常駐するデータオブジェクトとして、F D O、またはその F D O によって公開されるインターフェースは、アプリケーションプログラム 5 1 0 によって個別に認識される。

【 0 0 3 9 】

U S B ファンクションデバイスドライバ 5 2 5 は、プロキシ U S B バスドライバ 5 3 0 と通信する。プロキシ U S B バスドライバ 5 3 0 はデバイスに依存せず、ローカル U S B デバイスと、U S B デバイス 4 2 0 などのリモート U S B デバイスとを含む複数の U S B デバイスをサポートする。U S B ファンクションデバイスドライバ 5 2 5 は、P D O (physical device object) を作成する。この P D O は、U S B デバイス 4 2 0 の 1 つを表し、その U S B デバイス 4 2 0 の 1 つに固有のデータオブジェクトである。P D O データオブジェクトは、U S B デバイス 4 2 0 の 1 つを表し、その特定の U S B デバイスに固有のものである。一意または固有の P D O によって、サーバコンピュータ 4 0 5 は、プロキシ U S B バスドライバ 5 3 0 を使用して、特定の U S B デバイスを区別することができる。さらに、それぞれの中間インターフェース 5 0 0 および 5 0 5 を介して、プロキシ U S B バスドライバ 5 3 0 は、リモートクライアントアクセスドライバ 5 3 5 と通信する。リモートクライアントアクセスドライバ 5 3 5 は、サーバコンピュータ 4 0 5 から発生するコマンド (すなわち、アプリケーションプログラム 4 1 0 からのコマンド) を渡すという特定の機能を実行する。プロキシ U S B バスドライバ 5 3 0 は、U S B バスドライバ 5 1 5 と論理的に通信する。

【 0 0 4 0 】

アプリケーションプログラム 5 1 0 との通信は、I R P (I/O request packets) から、とりわけ I O C T L (I/O control) コードと呼ばれる特定のグループの I R P から構成することができる。I R P は、アプリケーションプログラム 5 1 0 から発生する通信データであり、データをアプリケーションプログラム 5 1 0 へ転送 (送信) するよう求める要求など、1 つまたは複数の U S B デバイス 4 2 0 にアクションを実行するよう求める要求を含む。I O C T L コードは、U S B ファンクションデバイスドライバ 5 2 5 と通信するために、アプリケーションプログラム 5 1 0 によって使用することができる。I O C T L コードは、特定のアクションを実行するために 1 つまたは複数の U S B デバイス 4 2 0 に対してアプリケーションプログラム 5 1 0 によって伝達される特定のコマンドまたは命令である。U S B ファンクションデバイスドライバ 5 2 5 は、アプリケーションプログラム 5 1 0 から I O C T L コードを受信し、その I O C T L コードを I R P に変換し、その I R P は、1 つまたは複数の U S B デバイス 4 2 0 へ渡される。同様に、1 つまたは複数の U S B デバイス 4 2 0 から送信される I R P は、I O C T L に変換され、この I O C T L は、アプリケーションプログラム 5 1 0 によって読み取ること、または使用することができる。I O C T L は、U S B ファンクションデバイスドライバ 5 2 5 などのデバイスま

10

20

30

40

50

たはクラスに固有のドライバによってアプリケーションプログラム 5 1 0 からのコマンドおよび/またはアクション(すなわち要求)を伝達するために、個別に使用される。USBファンクションデバイスドライバ 5 2 5 は、コマンドまたはアクションを定義する I O C T L を、特に I O C T L パラメータを検証する。

【 0 0 4 1 】

図 6 は、機能が、アプリケーションサーバコンピュータ 4 0 5 またはクライアントコンピュータ 4 1 0 のいずれかによって提供される、アプリケーションサーバコンピュータ 4 0 5 およびクライアントコンピュータ 4 1 0 における最上位のアーキテクチャの例を示している。1 つまたは複数の USB デバイス 4 2 0 は、クライアントコンピュータ 4 1 0 またはアプリケーションサーバコンピュータ 4 0 5 のいずれかによってアクセスおよび制御することができる。具体的には、アプリケーションサーバコンピュータ 4 0 5 またはクライアントコンピュータ 4 1 0 のいずれかにおけるアプリケーションプログラムが、特定の USB デバイス 4 2 0 に機能を提供する。

【 0 0 4 2 】

リモートクライアントアクセスシステムの実装形態では、クライアントコンピュータ 4 1 0 は主に、キーボードやマウスなどの入力デバイスを介してユーザによって入力されたデータなどの、入力および出力(I/O)を提供するために使用される。この例では、I/O データは、USB デバイス 4 2 0 によって提供される。クライアントコンピュータ 4 1 0 に常駐しているアプリケーションプログラム 6 0 5 を使用して、1 つまたは複数の USB デバイス 4 2 0 のための機能を提供することができ、これによって特に、I/O データを受信して、1 つまたは複数の USB デバイス 4 2 0 へ送信することができる。具体的には、1 つまたは複数の USB デバイス 4 2 0 によって提供されたデータまたは情報は、アプリケーションプログラム 6 0 5 を使用してクライアントコンピュータ 4 1 0 によってアクセスされる。アプリケーションプログラム 6 0 5 は、データまたは情報(すなわち制御)を特定の USB デバイス 4 2 0 へ提供することもできる。場合によっては、アプリケーションプログラム 6 0 5 などのアプリケーションプログラムがクライアントコンピュータ 4 1 0 に常駐しているか否かを問わず、1 つまたは複数の USB デバイス 4 2 0 に対するアクセスおよび制御を、サーバコンピュータ 4 0 5 に常駐しているアプリケーションプログラムへ導くか、または導き直すことが望ましい場合がある。アプリケーションプログラム 6 0 5 の例としては、ワードプロセッシングプログラム、マルチメディアプログラム、およびデータ管理プログラムなどがある。

【 0 0 4 3 】

アプリケーションプログラム 6 0 5 は、USB ファンクションデバイスドライバ 6 1 0 と通信する。USB ファンクションデバイスドライバ 6 1 0 は、USB デバイス 4 2 0 の 1 つがグループ化されているデバイスクラスに固有のものである。前述のように、特定の USB ファンクションデバイスドライバは、特定の USB デバイスをサポートする。たとえば USB ファンクションデバイスドライバは、デジタルカメラ、プリンタ、ディスクストレージユニット、スキャナなどに固有のものである。USB ファンクションデバイスドライバ 6 1 0 は、1 つまたは複数の USB デバイス 4 2 0 を表す F D O を作成し、同じクラスにグループ化されている USB デバイスは、同じ F D O によって表される。この特定の F D O は、1 つまたは複数の USB デバイス 4 2 0 がグループ化されているデバイスのクラスを表すデータオブジェクトであり、アプリケーションプログラム 6 0 5 へのインターフェースを公開または提供する。USB デバイス 4 2 0 がデジタルカメラである場合、作成される F D O (データオブジェクト)は、デジタルカメラ全般を表すものであり、特定のデジタルカメラを表すものではない。USB ファンクションデバイスドライバ 6 1 0 上に常駐するデータオブジェクトとして、F D O と、その F D O によって公開されるインターフェースは、アプリケーションプログラム 6 0 5、USB バスドライバ 5 1 5、および USB ファンクションデバイスドライバ 6 1 0 と通信する他のドライバ、モジュール、またはコンポーネントによって認識される。

【 0 0 4 4 】

10

20

30

40

50

USBファンクションデバイスドライバ610は、「ハブドライバ」としても知られているUSBバスドライバ515と通信する。USBバスドライバ515はデバイスに依存せず、USBデバイス420などの複数のUSBデバイスをサポートする。USBバスドライバ515は、USBデバイス420のうち特定のUSBデバイスを表すデータオブジェクトであるPDOを作成する。PDOデータオブジェクトの表示は、USBデバイス420のうち特定のUSBデバイスに固有のものである。たとえば、USBデバイス420が、接続されている複数のデジタルカメラの1つである場合、特定のUSBデバイスを識別するUSBハブドライバ515によって特定のPDOが作成される。このPDOは、USBホストコントローラ520へ個別に伝達される。さらに、USBバスドライバ515によって作成されたPDOを使用して、USBハブドライバ515と通信する他のドライバ、モジュール、またはコンポーネントに対して特定のUSBデバイスを識別することができる。

10

【0045】

USBホストコントローラ520は、1つまたは複数のUSBデバイス420への物理的なインターフェースとすることができ、USBバスドライバ515と通信する。USBホストコントローラ520は、USBポートに接続することができ、これらのUSBポートは、USBデバイス420へ接続される。1つまたは複数のUSBデバイス420がUSBホストコントローラ520へ接続されるたびに、USBホストコントローラ520は、接続された1つまたは複数のUSBデバイス420に常駐している識別子または識別データをUSBバスドライバ515へ渡す。USBバスドライバ515は、USBデバイス420を識別するPDOを作成し、このPDOは、他のドライバ、モジュール、またはコンポーネントに対して提示される。USBデバイス420のうち特定のUSBデバイスを識別することによって、クライアントコンピュータ410は（すなわち、クライアントコンピュータ410のCPUすなわちプロセッサは）、その特定のUSBデバイスにアクセスして使用する上で必要なソフトウェアをインストールする。これはPNP機能をサポートし、これによってユーザは、特定のUSBデバイスをサポートするための適切なソフトウェアをインストールする必要がなくなる。

20

【0046】

アプリケーションプログラム605との通信は、IRPから、とりわけIOCTLコードから構成することができる。IRPは、アプリケーションプログラム605から発生する通信データであり、データをアプリケーションプログラム605へ転送（送信）するよう求める要求など、1つまたは複数のUSBデバイス420にアクションを実行するよう求める要求を含む。IOCTLコードは、USBファンクションデバイスドライバ610と通信するために、アプリケーションプログラム605によって使用することができる。IOCTLコードは、特定のアクションを実行するためにアプリケーションプログラム605によって1つまたは複数のUSBデバイス420に対して伝達される、特定のコマンドまたは命令である。USBファンクションデバイスドライバ610は、アプリケーションプログラム605からIOCTLコードを受信し、そのIOCTLコードをIRPに変換し、そのIRPは、1つまたは複数のUSBデバイス420へ渡される。同様に、1つまたは複数のUSBデバイス420から送信されるIRPは、IOCTLに変換され、このIOCTLは、アプリケーションプログラム605によって読み取ること、または使用することができる。IOCTLは、USBファンクションデバイスドライバ610などのデバイスまたはクラスに固有のドライバによってアプリケーションプログラム605からのコマンドおよび/またはアクション（すなわち要求）を伝達するために、個別に使用される。USBファンクションデバイスドライバ610は、コマンドまたはアクションを定義するIOCTLを、特にIOCTLパラメータを検証する。

30

40

【0047】

クライアントトランスポートドライバ500が、クライアントコンピュータ410において提供され、これによって、アプリケーションサーバコンピュータ405とのリモートクライアントアクセス通信が可能となる。通信は、RDP通信プロトコルなどの通信プロ

50

トコルを実装することができ、具体的には、送信されるデータをコード化して、クライアントトランスポートドライバ500によって受信されたコマンドを復号するための通信プロトコルを実装することができる。さらにクライアントトランスポートドライバ500は、インターネットなどのネットワークを介してデータを通信する際のTCP/IPなどのトランスポートセッションプロトコルを実装することができる。

【0048】

場合によっては、1つまたは複数のUSBデバイス420をサポートする機能が、サーバコンピュータ405によって提供される。このような機能がアプリケーションサーバコンピュータ405によって提供される場合、リモートクライアントアクセスドライバ535がロードされる。リモートクライアントアクセスドライバ535は、アプリケーションサーバコンピュータ405から発生するコマンドを渡すことを含め、USBファンクションデバイスドライバ610によって提供される機能を実行する。リモートクライアントアクセスドライバ535を介して、USBデバイス420への通信は、アプリケーションプログラム605およびUSBファンクションデバイスドライバ610から導き直される。コマンドは、IRPの形態とすることができ、USBバスドライバ515へ渡すことができる。さらにリモートクライアントアクセスドライバ535は、USBバスドライバ515によって作成された特定のPDOを認識する。リモートクライアントアクセスドライバ535は、1つまたは複数のUSBデバイス420を表す特定のPDOと、その特定のPDOによって公開されるインターフェースとを明確に認識する。

【0049】

アプリケーションサーバコンピュータ405は、クライアントコンピュータ410などの1つまたは複数のクライアントコンピュータに特定の機能を提供する専用のアプリケーションソフトウェアを提供する。この例では、アプリケーションサーバコンピュータ405は、1つまたは複数のUSBデバイス420のための機能をサポートまたは提供するアプリケーションプログラム510を含む。アプリケーションプログラム510の例としては、ワードプロセッシングプログラム、マルチメディアプログラム、およびデータ管理プログラムなどがある。

【0050】

アプリケーションプログラム510は、USBファンクションデバイスドライバ525と通信する。USBファンクションデバイスドライバ525は、特定のUSBデバイス420がグループ化されているデバイスクラスに固有のものである。USBファンクションデバイスドライバ525は、USBデバイスの特定のクラスを表す特定のFDOを作成する。前述のように、この特定のFDOは、デバイスの1つのクラスを表すデータオブジェクトであり、アプリケーションプログラム510へのインターフェースを公開または提供する。USBファンクションデバイスドライバ525によって提供されるデータオブジェクトとして、FDOと、そのFDOによって公開されるインターフェースは、アプリケーションプログラム510によって個別に認識される。

【0051】

USBファンクションデバイスドライバ525は、プロキシUSBバスドライバ530と通信する。プロキシUSBバスドライバ530はデバイスに依存せず、USBデバイス420などの複数のローカルおよびリモートのUSBデバイスをサポートする。プロキシUSBバスドライバ530はPDOを作成する。このPDOは、特定のUSBデバイス420を表し、その特定のUSBデバイス420に固有のデータオブジェクトである。一意または固有のPDOによって、サーバコンピュータ405は、プロキシUSBバスドライバ530を使用して、特定のUSBデバイスを区別することができる。

【0052】

アプリケーションプログラム510との通信は、IRPおよびIOCTLコードから構成することができる。IOCTLコードは、USBファンクションデバイスドライバ525と通信するために、アプリケーションプログラム510によって個別に使用することができる。IOCTLコードは、アプリケーションプログラム510によって伝達される特

定のコマンドまたは命令である。USBファンクションデバイスドライバ525は、アプリケーションプログラム510からI/OCTLコードを受信し、そのI/OCTLコードをIRPに変換し、そのIRPは、プロキシUSBバスドライバ530へ渡される。プロキシUSBバスドライバ530によって受信されるIRPは、サーバトランスポートドライバ505によって受信される。サーバトランスポートドライバ505はまた、プロキシUSBバスドライバ530にIRPを渡すことができ、このようなIRPは、USBデバイス420などのリモートUSBデバイスからのデータおよび要求を表す。

【0053】

図7は、USBデバイスに対するアクセスおよび制御をアプリケーションサーバコンピュータ405などのリモートコンピュータへ導く、または導き直すプロセス700を示している。プロセス700は、論理的なフローグラフ内のブロックの集合として示されており、ハードウェア、ソフトウェア、ファームウェア、またはそれらの組合せにおいて実施できる一連のオペレーションを表している。ソフトウェアのコンテキストでは、それらのブロックは、1つまたは複数のプロセッサによって実行されると、列挙されているオペレーションを実行する、コンピュータ命令を表す。プロセス700については、図4、5、および6に記載のアプリケーションサーバコンピュータ405およびクライアントコンピュータ410を参照して説明する。ここでは、アプリケーションサーバコンピュータ405およびクライアントコンピュータ410が、特定のプロセスを実行するための手段を提供する。フローチャートとして説明するが、これらのプロセスは同時に発生できることを意図している。

【0054】

ブロック705において、USBデバイスはクライアントコンピュータに接続される。このデバイスは、クライアントコンピュータ上のUSBポートへ物理的に接続することができる。USBポートは、USBコントローラと物理的に接続される。USBバスドライバは、USBコントローラと論理的に接続される。USBバスドライバは、特定のUSBデバイスを認識および識別し、そのUSBデバイスがグループ化されているクラスをさらに識別する。この認識プロセスは、この例ではUSBバスドライバによって個別に実行され、この認識プロセスを介して、USBデバイスのPNP (plug and play) が提供される。すなわち、USBデバイスが認識され、該当する場合は、適切なソフトウェアがユーザの介入を伴わずにインストールされる。PDO (physical device object) と呼ばれるデータオブジェクトと、そのUSBデバイスに固有のインターフェースとが、USBバスドライバによって個別に提供される。このPDOおよびインターフェースは、USBバスドライバに接続されているコンポーネント、モジュール、およびドライバに伝達される。

【0055】

場合によっては、接続されているUSBデバイスに対する機能、またはアクセスおよび制御をクライアントコンピュータが提供すること（すなわち、ブロック710の「いいえ」の分岐に進むこと）が望ましい。ブロック715では、クライアントコンピュータにおけるアプリケーションプログラムが、そのUSBデバイスのための機能を提供する場合、特定のUSBファンクションデバイスドライバがインストールまたはロードされる。USBファンクションデバイスドライバは、そのUSBデバイスのデバイスクラスに固有のものである。USBファンクションデバイスドライバは、FDO (functional device object) と、接続されているUSBデバイスに固有のインターフェースとを個別に提供する。このFDOおよびインターフェースは、USBファンクションドライバに接続されているコンポーネント、モジュール、およびドライバに伝達される。アプリケーションプログラムからUSBデバイスへの通信パスが提供される。

【0056】

ブロック720では、USBデバイスとアプリケーションプログラムの間で通信が実行される。アプリケーションプログラムは、データを求める要求について記述するI/OCTL (I/O control) コードを送信することができる。アプリケーションプログラムは、I/OCTLコードを受信することもできる。I/OCTLコードは、アプリケーションプログ

10

20

30

40

50

ラムから送信された後で I R P (I/O request packets) に変換することができる。さらに、アプリケーションプログラムによって受信された I O C T L コードは、U S B デバイスからの I R P として開始することができる。

【 0 0 5 7 】

他の特定の場合には、接続されている U S B デバイスに対する機能、またはアクセスおよび制御をアプリケーションサーバコンピュータが提供すること（すなわち、ブロック 7 1 0 の「はい」の分岐に進むこと）が望ましい。典型的なケースとしては、クライアントコンピュータが、適切な機能を U S B デバイスに提供する上で限られたアプリケーションプログラムしか有していない状況や、そうしたアプリケーションプログラムをまったく有していない状況などがある。他のケースでは、アプリケーションプログラムがクライアントコンピュータに常駐して、U S B デバイスのための機能を提供することができるにもかかわらず、クライアントコンピュータにおけるユーザが、U S B デバイスのための機能をアプリケーションサーバコンピュータから提供されたいと希望する場合がある。

【 0 0 5 8 】

ブロック 7 2 5 では、リモートクライアントアクセスドライバが、クライアントコンピュータにおいてインストールまたはロードされる。アプリケーションサーバコンピュータからの通信、特にサーバコンピュータにおけるアプリケーションプログラムは、リモートクライアントアクセスドライバを介して導かれ、基本的にアプリケーションサーバコンピュータ（すなわち、アプリケーションサーバコンピュータのアプリケーションプログラム）から U S B デバイスへの通信パスを提供する。クライアントトランスポートドライバは、クライアントコンピュータにおいて提供することができ、これによってアプリケーションサーバコンピュータとの通信が可能となる。アプリケーションサーバコンピュータは、クライアントコンピュータのクライアントトランスポートドライバへの通信を受信および送信するサーバトランスポートドライバを含むことができる。

【 0 0 5 9 】

ブロック 7 3 0 では、アプリケーションサーバコンピュータにおいて U S B デバイスが認識される。詳細には、アプリケーションサーバコンピュータに実装されているプロキシ U S B バスドライバが、U S B デバイスと、その U S B デバイスの特定のクラスとを認識する。

【 0 0 6 0 】

ブロック 7 3 5 では、リモートクライアントアクセスバスドライバが、U S B デバイスのクラスに固有の U S B ファンクションデバイスドライバをロードまたはインストールする。U S B ファンクションデバイスドライバは、その U S B デバイスが属するデバイスクラスまたはグループに固有のものである。

【 0 0 6 1 】

ブロック 7 4 0 では、U S B デバイスと、アプリケーションサーバコンピュータに常駐しているアプリケーションプログラムの間で通信が実行される。アプリケーションプログラムは、データを求める要求について記述する I O C T L を送信することができ、これらの I O C T L は I R P に変換され、それらの I R P は U S B デバイスによって受信される。U S B ファンクションデバイスドライバによって、F D O（デバイスに固有のデータオブジェクト）を作成し、提供することができる。この F D O は、そのアプリケーションプログラムにとっての U S B デバイスを表す。

【 0 0 6 2 】

（典型的なコンピュータ）

図 8 は、この主題の態様を実施するための環境として、たとえばクライアントコンピュータ 4 1 0 またはアプリケーションサーバコンピュータ 4 0 5 として適切な典型的なコンピュータ 8 0 0 を示している。コンピュータ 8 0 0 のコンポーネントは、処理装置 8 2 0 と、システムメモリ 8 3 0 と、システムメモリ 8 3 0 を含むさまざまなシステムコンポーネントを処理装置 8 2 0 に結合するシステムバス 8 2 1 とを含むことができるが、これらには限定されない。システムバス 8 2 1 は、メモリバ

スまたはメモリコントローラと、ペリフェラルバスと、さまざまなバスアーキテクチャのいずれかを使用するローカルバスとを含む複数のタイプのバス構造のいずれにすることもできる。たとえばこのようなアーキテクチャは、I S A (Industry Standard Architecture) バス、M C A (Micro Channel Architecture) バス、E I S A (Enhanced ISA) バス、V E S A (Video Electronics Standards Association) ローカルバス、およびメザニンバスとしても知られているP C I (Peripheral Component Interconnect) バスを含むが、これらには限定されない。

【0063】

典型的なコンピュータ800は通常、さまざまなコンピュータ読取り可能記憶媒体を含む。コンピュータ読取り可能記憶媒体は、コンピュータ800によってアクセスできる利用可能な任意のメディアとすることができ、揮発性および不揮発性記憶媒体、ならびに取外し可能および取外し不可能な記憶媒体の双方を含む。たとえばコンピューティングデバイス読取り可能記憶媒体は、コンピュータ記憶媒体および通信媒体を含むことができるが、これらには限定されない。コンピュータ記憶媒体は、コンピュータ読取り可能命令、データ構造、プログラムモジュール、他のデータなどの情報を記憶するための任意の方法または技術において実装される揮発性および不揮発性記憶媒体、ならびに取外し可能および取外し不可能な記憶媒体を含む。コンピュータ記憶媒体は、R A M、R O M、E E P R O M、フラッシュメモリまたは他のメモリ技術、C D - R O M、デジタル多用途ディスク(D V D)または他の光ディスク記憶装置、磁気カセット、磁気テープ、磁気ディスク記憶装置または他の磁気記憶装置、あるいは希望の情報を保存するために使用可能で、コンピューティングデバイス800によってアクセス可能な他の任意の媒体を含むが、これらには限定されない。通信媒体は通常、搬送波や他の伝送メカニズムなどの被変調データ信号内のコンピュータ読取り可能命令、データ構造、プログラムモジュール、または他のデータを具体化し、任意の情報伝達媒体を含む。「被変調データ信号」という用語は、情報をその信号内でコード化するような方法で設定または変更されたその特性のうちの1つまたは複数を有する信号を意味する。たとえば通信媒体は、有線ネットワークや直接有線接続などの有線媒体と、音響、無線周波数(R F)、赤外線、他の無線媒体などの無線媒体とを含むが、これらには限定されない。また上記のいずれの組合せも、コンピューティングデバイス読取り可能記憶媒体の範囲内に含まれるものである。

【0064】

システムメモリ830はコンピューティングデバイス記憶媒体を読み取り専用メモリ(R O M)831およびランダムアクセスメモリ(R A M)832などの揮発性メモリおよび/または不揮発性メモリの形態で含む。基本入出力システム833(B I O S)は、起動中などにコンピュータ800内の要素間における情報伝達を補助する基本ルーチンを含み、通常はR O M831内に格納されている。R A M832は通常、処理装置820がすぐにアクセスできるか、および/または処理装置820によってその時点で操作されているデータモジュールおよび/またはプログラムモジュールを含む。図8は、例としてオペレーティングシステム834、アプリケーションプログラム835、その他のプログラムモジュール836、およびプログラムデータ837を示しているが、これらには限定されない。

【0065】

また典型的なコンピュータ800は、他の取外し可能/取外し不可能、揮発性/不揮発性コンピュータ記憶媒体を含むこともできる。図8は、例示のみを目的として、取外し不可能な不揮発性の磁気記憶媒体との間で読み取りや書き込みを行うハードディスクドライブ841と、取外し可能な不揮発性の磁気ディスク852との間で読み取りや書き込みを行う磁気ディスクドライブ851と、C D - R O Mや他の光メディアなどの取外し可能な不揮発性の光ディスク856との間で読み取りや書き込みを行う光ディスクドライブ855とを示している。典型的な動作環境において使用できる他の取外し可能/取外し不可能、揮発性/不揮発性コンピューティングデバイス記憶装置としては、磁気テープカセット、フラッシュメモリカード、デジタル多用途ディスク、デジタルビデオテープ、ソリッド

10

20

30

40

50

ステートRAM、ソリッドステートROMなどがあるが、これらには限定されない。ハードディスクドライブ841は通常、インターフェース840などの取外し不可能なメモリインターフェースを介してシステムバス821に接続されており、磁気ディスクドライブ851および光ディスクドライブ855は通常、インターフェース850などの取外し可能なメモリインターフェースによってシステムバス821に接続されている。

【0066】

図8に示した上述のドライブおよびそれらに関連するコンピューティングデバイス記憶装置は、コンピュータ800用のコンピュータ読取り可能命令、データ構造、プログラムモジュール、および他のデータの記憶を提供する。たとえば図8において、ハードディスクドライブ841は、オペレーティングシステム844、アプリケーションプログラム845、その他のプログラムモジュール846、およびプログラムデータ847を記憶するものとして図示されている。これらのコンポーネントは、オペレーティングシステム834、アプリケーションプログラム835、その他のプログラムモジュール836、およびプログラムデータ837と同じであっても、または異なってもよいという点に留意されたい。ここでは、オペレーティングシステム844、アプリケーションプログラム845、その他のプログラムモジュール846、およびプログラムデータ847が最低限異なるコピーであることを示すために、異なる番号を割り当てている。ユーザは、キーボード848、および通常はマウス、トラックボール、またはタッチパッドと呼ばれるポインティングデバイス861などの入力デバイスを介して典型的なコンピュータ800にコマンドおよび情報を入力することができる。他の入力デバイス(図示せず)は、マイクロフォン、ジョイスティック、ゲームパッド、衛星放送受信アンテナ、スキャナなどを含むことができる。これらの入力デバイスおよび他の入力デバイスは、システムバスに結合されているユーザ入力インターフェース860を介して処理装置820に接続される場合が多いが、パラレルポート、ゲームポート、あるいはとりわけUSBポートなどの他のインターフェースおよびバス構造によって接続することもできる。またモニター862や他のタイプの表示装置も、ビデオインターフェース890などのインターフェースを介してシステムバス821に接続される。コンピューティングデバイスは、モニター862に加えて、スピーカ897およびプリンタ896などの他の周辺出力デバイスを含むこともでき、これらは周辺出力インターフェース895を介して接続することができる。

【0067】

典型的なコンピュータ800は、リモートコンピューティングデバイス880などの1つまたは複数のリモートコンピューティングデバイスへの論理接続を使用して、ネットワーク化された環境内で動作することができる。リモートコンピューティングデバイス880は、パーソナルコンピューティングデバイス、サーバ、ルータ、ネットワークPC、ピアデバイス、または他の一般的なネットワークノードとすることができ、図8にはメモリ記憶装置881しか示されていないが、通常はコンピュータ800に関連する上述の要素の多くまたはすべてを含む。図8に示されている論理接続は、ローカルエリアネットワーク(LAN)871およびワイドエリアネットワーク(WAN)873を含むが、他のネットワークを含むこともできる。こうしたネットワーキング環境は、オフィス、企業規模のコンピューティングデバイスネットワーク、イントラネット、およびインターネットにおいてよく見受けられる。

【0068】

LANネットワーキング環境において使用する場合、典型的なコンピュータ800は、ネットワークインターフェースまたはアダプタ884を介してLAN871に接続される。WANネットワーキング環境において使用する場合、典型的なコンピュータ800は通常、モデム872、またはインターネットなどのWAN873上で通信を確立するための他の手段を含む。モデム872は内蔵型または外付け型とすることができ、ユーザ入力インターフェース860または他の適切なメカニズムを介してシステムバス821に接続することができる。ネットワーク化された環境では、典型的なコンピュータ800に関連して示されているプログラムモジュール、またはその一部をリモートメモリ記憶装置内に格

10

20

30

40

50

納することができる。図 8 は、例としてリモートアプリケーションプログラム 885 をメモリデバイス 881 上に常駐するものとして示しているが、この形態には限定されない。示されているネットワーク接続は代表的なものであり、コンピューティングデバイス間に通信リンクを確立する他の手段も使用できることが理解できるであろう。

【0069】

(結論)

クライアントコンピュータに接続されている USB デバイスに対するアクセスおよび制御をサーバコンピュータへ選択的に導くまたは導き直すプロセスについて上述した。本発明について、構造的な特徴および/または方法論的な行為に特有の言葉で説明したが、添付の特許請求の範囲において定義される本発明は、説明した特定の特徴または行為に必ずしも限定されるものではないことを理解されたい。むしろ特定の特徴および行為は、特許請求された本発明を実施する典型的な形態として開示されている。

【図面の簡単な説明】

【0070】

【図 1】1 台のコンピュータにおいて実装される従来技術の USB アーキテクチャを示すブロック図である。

【図 2】複数のアプリケーションプログラムおよび複数の USB デバイスと共に 1 台のコンピュータにおいて実装される従来技術の USB アーキテクチャを示すブロック図である。

【図 3】1 つまたは複数の USB デバイスのリモートアクセスの従来技術によるアーキテクチャを示すブロック図である。

【図 4】1 つまたは複数の USB デバイスがクライアントコンピュータまたはアプリケーションサーバコンピュータのいずれかによって制御されるリモートクライアントアクセスシステムを示す図である。

【図 5】USB デバイスのための機能がアプリケーションサーバコンピュータにおいて提供される、クライアントコンピュータおよびアプリケーションサーバコンピュータを示すブロック図である。

【図 6】USB デバイスにアクセスして制御するクライアントコンピュータおよびアプリケーションサーバコンピュータを示すブロック図である。

【図 7】クライアントコンピュータの USB デバイスに対する制御を、クライアントコンピュータまたはアプリケーションサーバコンピュータへ導くまたは導き直すプロセスを示す流れ図である。

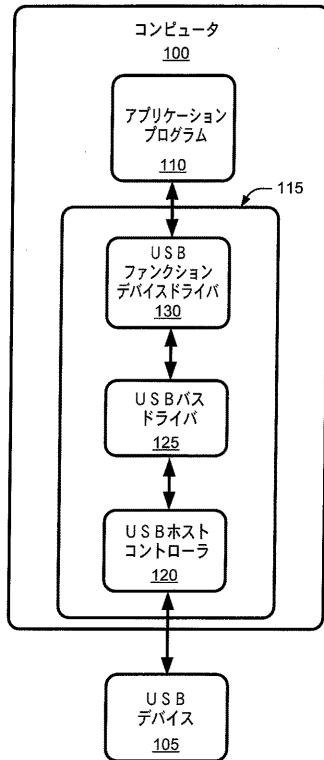
【図 8】クライアントコンピュータまたはアプリケーションサーバコンピュータの実装形態を示すブロック図である。

10

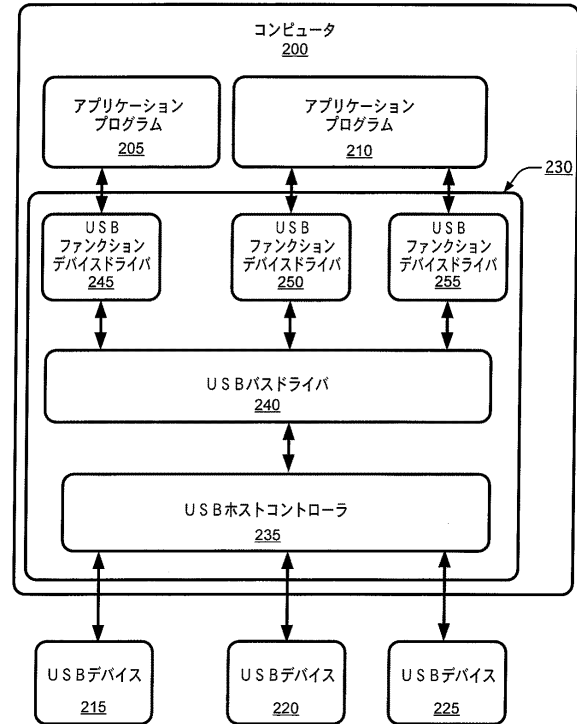
20

30

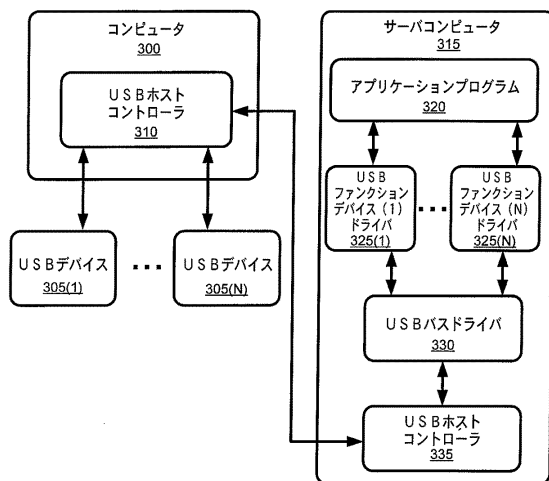
【図 1】



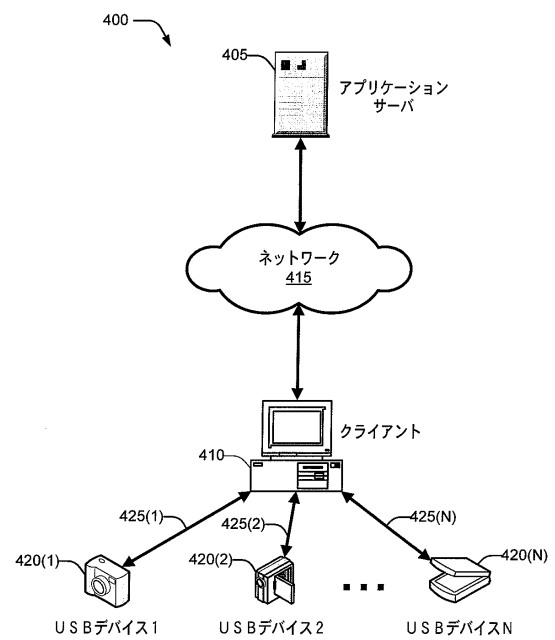
【図 2】



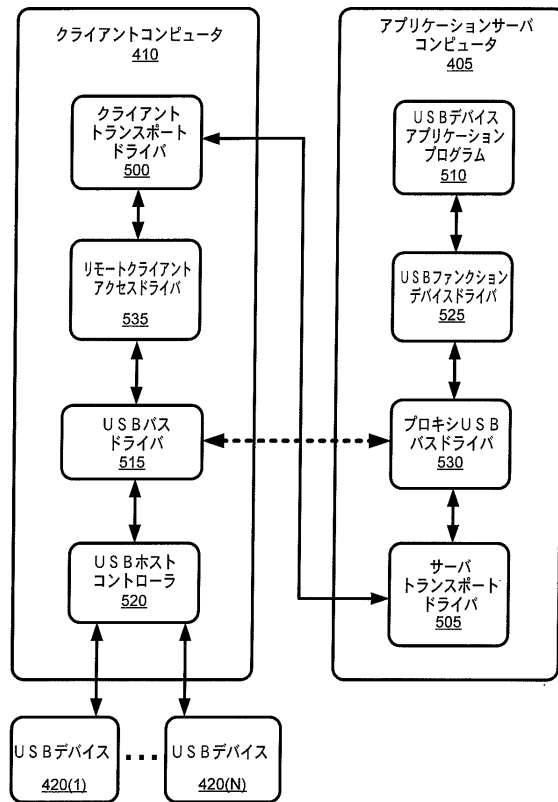
【図 3】



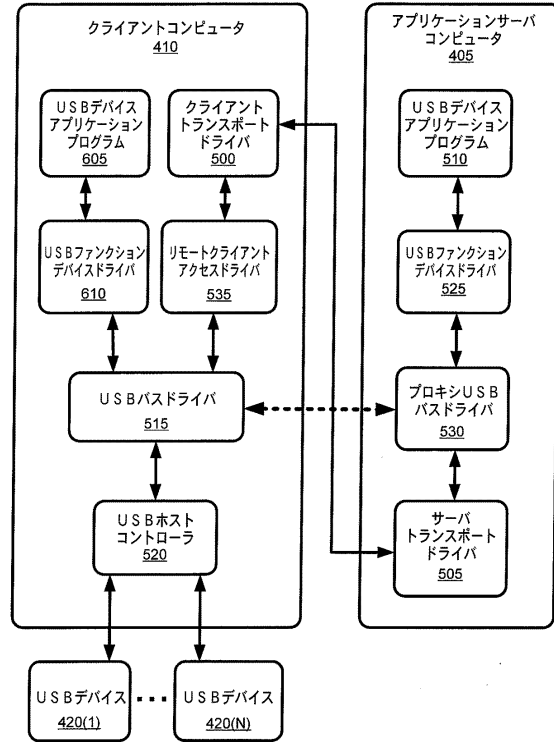
【図 4】



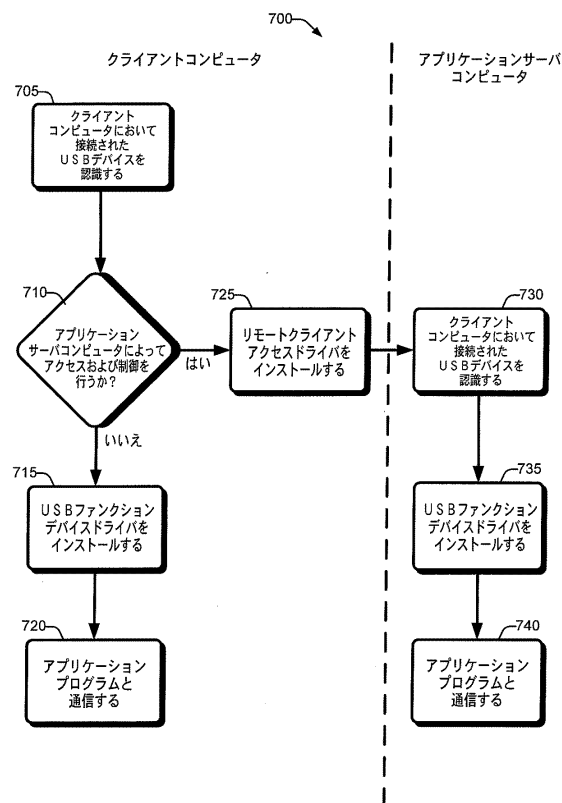
【 図 5 】



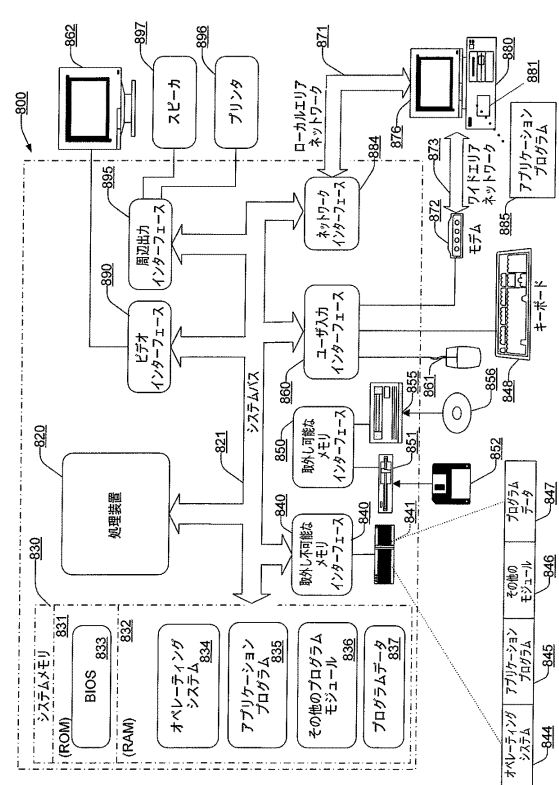
【 図 6 】



【 図 7 】



【 図 8 】



フロントページの続き

- (72)発明者 チェンイン チョン
アメリカ合衆国 98052 ワシントン州 レッドモンド ワン マイクロソフト ウェイ マイ
クロソフト コーポレーション内
- (72)発明者 ジョン シー・ダン
アメリカ合衆国 98052 ワシントン州 レッドモンド ワン マイクロソフト ウェイ マ
イクロソフト コーポレーション内
- (72)発明者 ジョイ チック
アメリカ合衆国 98052 ワシントン州 レッドモンド ワン マイクロソフト ウェイ マ
イクロソフト コーポレーション内
- (72)発明者 マカランド ヴィ・パトワーダン
アメリカ合衆国 98052 ワシントン州 レッドモンド ワン マイクロソフト ウェイ マ
イクロソフト コーポレーション内
- (72)発明者 ビノッド エム・マムタニ
アメリカ合衆国 98052 ワシントン州 レッドモンド ワン マイクロソフト ウェイ マ
イクロソフト コーポレーション内

審査官 横山 佳弘

- (56)参考文献 特開平11-184793(JP,A)
特開2000-251012(JP,A)
国際公開第2001/025934(WO,A1)
特開平11-007404(JP,A)
特開2004-172902(JP,A)
特表2004-527817(JP,A)

- (58)調査した分野(Int.Cl., DB名)
G06F 13/38
G06F 13/10
G06F 13/00