

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第4923053号  
(P4923053)

(45) 発行日 平成24年4月25日 (2012. 4. 25)

(24) 登録日 平成24年2月10日 (2012. 2. 10)

(51) Int. Cl.

F I

G 0 6 F 21/24 (2006. 01)

G 0 6 F 21/24 1 6 0 A

G 0 6 F 3/14 (2006. 01)

G 0 6 F 3/14 3 5 0 A

G 0 6 T 1/00 (2006. 01)

G 0 6 T 1/00 A

請求項の数 12 (全 14 頁)

(21) 出願番号 特願2008-531220 (P2008-531220)  
 (86) (22) 出願日 平成18年9月11日 (2006. 9. 11)  
 (65) 公表番号 特表2009-508262 (P2009-508262A)  
 (43) 公表日 平成21年2月26日 (2009. 2. 26)  
 (86) 国際出願番号 PCT/US2006/035334  
 (87) 国際公開番号 W02007/033088  
 (87) 国際公開日 平成19年3月22日 (2007. 3. 22)  
 審査請求日 平成21年9月11日 (2009. 9. 11)  
 (31) 優先権主張番号 60/716, 364  
 (32) 優先日 平成17年9月12日 (2005. 9. 12)  
 (33) 優先権主張国 米国 (US)  
 (31) 優先権主張番号 11/304, 299  
 (32) 優先日 平成17年12月15日 (2005. 12. 15)  
 (33) 優先権主張国 米国 (US)

(73) 特許権者 500046438  
 マイクロソフト コーポレーション  
 アメリカ合衆国 ワシントン州 9805  
 2-6399 レッドモンド ワン マイ  
 クロソフト ウェイ  
 (74) 代理人 100077481  
 弁理士 谷 義一  
 (74) 代理人 100088915  
 弁理士 阿部 和夫  
 (72) 発明者 ウィルヘルム シュミーター  
 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

最終頁に続く

(54) 【発明の名称】 リモートクライアントのディスプレイからの隠しデータをフィルタリングする方法

(57) 【特許請求の範囲】

【請求項 1】

送信側コンピュータシステムが、共有アプリケーションの描画コマンドと隠しアプリケーションの描画コマンドとをフィルタリングするための方法であって、前記送信側コンピュータシステムは、受信側コンピュータシステムとネットワーク接続を介してデスクトップ表示を共有するように構成され、前記方法は、

前記送信側コンピュータシステムと前記受信側コンピュータシステムの両方で表示されるように意図された1つまたは複数の共有描画コマンドを識別する動作と、

前記送信側コンピュータシステムで表示されるように意図されているが前記受信側コンピュータシステムから隠されるように意図された1つまたは複数の隠し描画コマンドを識別する動作と、

前記送信側コンピュータシステムのドライバインターフェイスが前記共有描画コマンドと前記隠し描画コマンドの両方を受信する動作と、

前記ドライバインターフェイスが、前記送信側コンピュータシステムのディスプレイドライバを識別する動作と、

前記ドライバインターフェイスが前記ディスプレイドライバに前記共有描画コマンドと前記隠し描画コマンドを送信する動作と、

前記ディスプレイドライバが前記共有描画コマンドと前記隠し描画コマンドをレンダリングする動作と、

前記レンダリングされた共有描画コマンドと隠し描画コマンドを前記送信側コンピュ

10

20

タシステムのローカルディスプレイ装置に表示する動作と、

前記ドライバインターフェイスが、リモートビュードライバである、前記送信側コンピュータシステムのミラードライバを識別する動作と、

前記ドライバインターフェイスが前記ミラードライバに前記共有描画コマンドと前記隠し描画コマンドを送信する動作と、

前記ミラードライバが前記共有描画コマンドを、前記送信側コンピュータシステムの送信側ビットマップにレンダリングする動作と、

前記ミラードライバが、前記隠し描画コマンドの代わりに代替イメージを前記受信側コンピュータシステムに送信できるように、前記隠し描画コマンドの代替イメージを前記送信側ビットマップにレンダリングする動作と、

前記レンダリングされた共有描画コマンドと前記レンダリングされた代替イメージとを含むネットワークパケットを、リモートディスプレイ装置を有する前記受信側コンピュータシステムへ送信する動作と

を含むことを特徴とする方法。

【請求項 2】

前記ミラードライバが、前記代替イメージを受信して、少なくとも 1 つの隠し描画コマンドを該代替イメージに置き換える動作をさらに含むことを特徴とする請求項 1 に記載の方法。

【請求項 3】

前記ドライバインターフェイスと前記ミラードライバはカーネルモードで動作することを特徴とする請求項 1 に記載の方法。

【請求項 4】

前記ディスプレイドライバが前記共有描画コマンドと前記隠し描画コマンドをレンダリングする動作は、前記共有描画コマンドの各々と前記隠し描画コマンドの各々を前記ローカルディスプレイ装置にレンダリングすることを含むことを特徴とする請求項 1 に記載の方法。

【請求項 5】

前記ドライバインターフェイスが、前記隠し描画コマンドを生成した隠しアプリケーションから提供された前記代替イメージを、前記ミラードライバへ送信する動作をさらに含むことを特徴とする請求項 1 に記載の方法。

【請求項 6】

前記ミラードライバは、リモートデスクトッププロトコルドライバであり、前記ネットワークパケットを前記受信側コンピュータシステムに送信する動作は、前記ネットワークパケットを、リモートデスクトッププロトコルを通して前記受信側コンピュータシステムに送信することを含むことを特徴とする請求項 1 に記載の方法。

【請求項 7】

前記ネットワークパケットは、前記送信側コンピュータシステムから前記受信側コンピュータシステムに送信されると、前記共有描画コマンドが表示され、前記隠し描画コマンドが前記代替イメージによって描写されるように、前記受信側コンピュータシステムにおいて表示用にレンダリングされることを特徴とする請求項 6 に記載の方法。

【請求項 8】

前記ミラードライバが、前記共有描画コマンドあるいは前記隠し描画コマンドのいずれかを生成したアプリケーションからの停止命令を受信する動作と、

前記共有描画コマンドおよび前記隠し描画コマンドのいずれかの少なくとも一部を停止イメージファイルで代用する動作と

をさらに含むことを特徴とする請求項 1 に記載の方法。

【請求項 9】

前記ミラードライバが前記アプリケーションからの前記停止命令を受信し、前記ミラードライバが前記共有描画コマンドあるいは前記隠し描画コマンドのいずれかの代わりに前記停止イメージファイルを前記送信側ビットマップにレンダリングすることを特徴とする

10

20

30

40

50

請求項 8 に記載の方法。

【請求項 10】

前記ドライバインターフェイスが、前記ミラードライバに前記共有描画コマンドと前記隠し描画コマンドを送信する動作は、前記ドライバインターフェイスが、前記ミラードライバに、前記共有描画コマンドと前記隠し描画コマンドとともにフィルタリング命令を送信することを含み、

前記フィルタリング命令は、前記隠し描画コマンドの代替イメージを作成するための幾何学的な命令と、前記隠し描画コマンドを共有しないことの指示を含むことを特徴とする請求項 1 に記載の方法。

【請求項 11】

前記ドライバインターフェイスが前記ミラードライバに前記共有描画コマンドと前記隠し描画コマンドを送信する動作は、前記共有描画コマンドが、前記受信側コンピュータシステムと共有される共有ディスプレイ画面の特定の領域に関連するかどうかを判断し、前記共有描画コマンドが前記共有ディスプレイ画面の前記特定の領域に関連する場合にのみ、前記共有描画コマンドを前記ミラードライバに送信することを特徴とする請求項 10 に記載の方法。

【請求項 12】

ネットワーク接続を介してデスクトップの表示を受信側コンピュータシステムと共有するように構成され、少なくともドライバインターフェイスと、ディスプレイドライバと、ミラードライバと、ディスプレイ装置とを備えた送信側コンピュータシステムのプロセッサに、請求項 1 に記載の方法を実行させることを特徴とするコンピュータプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、リモートクライアントのディスプレイからの隠し ( o b s c u r e d ) データをフィルタリングする ( f i l t e r i n g ) 方法に関する。

【背景技術】

【0002】

コンピュータ化されたシステムは、ますます普及しており、このため規模の大小を問わないネットワーク内でコンピュータシステムのファイルと処理リソース ( 資源 ) を分散する必要がある。一般に、コンピュータシステムと関連のデバイスは、たとえば個人的な電子メッセージの交換、商品の販売、アカウント情報の提示など、さまざまな理由でネットワークを経由して情報を伝達する。しかし、コンピュータシステムとその関連のアプリケーションはますます機能が向上しているので、ネットワーク上のデータとリソースの共有に関連する問題も増えていることがわかるであろう。

【0003】

組織的なネットワーク内でリソースを分散する現在のいくつかの方法には、集中コンピューティング ( c e n t r a l i z e d   c o m p u t i n g ) のシナリオが含まれており、その集中コンピューティングはそのようなソースをローカルにインストールしていない 1 つまたは複数のクライアントとリソースを共有する集中ネットワークサーバー ( c e n t r a l i z e d   n e t w o r k   s e r v e r ) を含んでいる可能性がある。このような機能のために使用されている当該プロトコルは、リモートデスクトッププロトコル ( " R D P " : R e m o t e   D e s k t o p   P r o t o c o l ) である。リモートデスクトッププロトコルを使用すると、クライアントコンピュータシステムは対象のリソースをホストする集中ネットワークサーバーにアクセスできる。クライアントコンピュータシステムは、ちょうどこうしたリソースがあたかもローカルにインストールされているかのように、こうしたリソースと交信する (たとえば、マウスイベントやキーボードイベントなどを送信する)。

【0004】

次に、ネットワークサーバーはこうした対話を処理し、サーバー自体のビデオドライバ

10

20

30

40

50

を使用してデータの対応するレンダリング情報を作成し、処理されたデータと作成されたレンダリング情報の両方をクライアントに返送する。そして、クライアントコンピュータシステムはこうしたデータとレンダリング情報を受信し、クライアント側のビデオドライバを使用して受信したデータをローカルにレンダリングし、表示する。こうしたクライアントコンピュータシステムとネットワークサーバーとのこの対話は、あたかもクライアントコンピュータシステムがそれ自体にインストールされたリソースを使用して実際にデータをローカルに処理しているかのように途切れなく（シームレスに）実行されるのが理想的である。残念ながら、こうしたシステムはネットワークスループット（処理能力、処理量）の制約による影響を受けやすく、結果として、負荷が大きい場合は対話と処理の点からローカルクライアントコンピュータシステムでの表示間で「ラグ（lag）」（通信の時間差による遅れ）となる可能性がある。さらに、こうしたシステムにはサーバーで表示されるいくつかのウィンドウは、あるローカルクライアントでも表示されるが他のローカルウィンドウでは表示されないというような、「共有（shared）」リソースと「非共有（unshared）」リソースのコンセプト（概念）を有していないのが一般的である。むしろ、クライアントセッションに関してネットワークサーバーで開かれているすべては、クライアントコンピュータシステムに配信することができるものである。

10

#### 【0005】

いくつかの点で前述の集中共有モデル（centralized sharing model）と同様である別のタイプのシステムには、ネットワーク上の他の受信クライアントコンピュータシステムにウィンドウデータの情報を送信するように設定されたブロードキャスト（または「送信側」）クライアントコンピュータシステムが含まれる。この機能は、「デスクトップ共有（desktop sharing）」と呼ばれることもある。この例では、ブロードキャストコンピュータ（たとえば学習環境の「インストラクタ」）と受信側コンピュータシステム（たとえば「受講生」）は、普通にインストールされているアプリケーションプログラムを使用して接続し、インストラクタ用コンピュータのデスクトップの表示とローカルにインストールされたアプリケーションとを共有できるようにする。集中コンピューティングシステムのシナリオと同様に、クライアントのコンピュータシステムは、インストラクタのコンピュータで表示されるウィンドウがあたかも受講生のコンピュータ自体のウィンドウであるかのように、それらウィンドウで情報のやりとりができる可能性がある。

20

30

#### 【0006】

前述のRDPを使用したクライアント-サーバーモデルとは対照的に、こうしたシステムでは共有ウィンドウデータと非共有ウィンドウデータの概念を実施できる。たとえば、インストラクタはインストラクタが開いているウィンドウのすべてを受講生と共有したいとは限らない場合（たとえば、インターネットブラウザが銀行口座を開いている場合）があるが、このような場合は、解答を含むウィンドウは共有するが、機密情報を含むウィンドウは共有しないように指定できる。とは言うものの、インストラクタのコンピュータシステムは非共有ウィンドウからのデータをそれでも暗号化（encode）してから、共有ウィンドウデータと非共有ウィンドウデータの両方を受講生のコンピュータシステムに送信することになる。その結果として、受講生のコンピュータシステムで共有ウィンドウは正確にレンダリングされるものの、非共有ウィンドウのデータはそれが独自に暗号化されているため、処理しようとしてもレンダリングできないこととなる。それ故に、受講生のコンピュータシステムでは、本来非共有ウィンドウデータが表示されるはずの場所に、いくつかのデフォルトの一般的なラインや正方形が生成されることになる。

40

#### 【発明の開示】

#### 【発明が解決しようとする課題】

#### 【0007】

前述の集中RDPモデルと同様に、このインストラクタ/受講生（すなわち送信機/受信機）タイプのシステムにもいくつかの欠点があるため、集中デスクトップ共有環境への適用は困難になっている。たとえば、共有概念と非共有概念が適切に管理されるように、

50

同じアプリケーションプログラムと対応する更新をローカルにインストールするためには、こうしたシステムでは典型的に送信プログラムと受信プログラムの両方を必要としている。さらに、非共有データを暗号化した場合に、受信側コンピュータが非共有ウィンドウデータの復号化を試行し、レンダリングできない場合に表示するものを決定するということが必要としていることは、一定の効率の悪さがある。さらに、非共有データを送信する場合は、暗号化されていても、悪徳 ( u n s c r u p u l o u s ) エージェントがこの非共有データを取得し、復号化に成功する場合がないとは言い切れないという可能性が依然として存在しているので、セキュリティ上の問題が存在し得る。

#### 【 0 0 0 8 】

それに対して、共有ウィンドウデータと非共有ウィンドウデータの場合を考慮する場合は、RDPなどのようなプロトコルを更新するだけでも簡単なことではない。具体的に言うと、こうしたプロトコルを更新することは、一般的に、プロトコルを介してサーバーからのリソースを共有する、サーバーと各クライアントコンピュータシステムの両方において対応する拡張を提供しインストールすることに、アドミニストレータ ( 運営管理者 ) が時間をかける必要があることを含んでいる。

#### 【 0 0 0 9 】

したがって、現在のリソース共有プロトコルにはさまざまな側面 ( a s p e c t ) があるため、デスクトップ共有環境でこうしたプロトコルの利点を組み合わせることを困難にしている。

#### 【 課題を解決するための手段 】

#### 【 0 0 1 0 】

本発明の実施により、ネットワーク化されたコンピュータシステム間でアプリケーションの表示の共有についての効率を高めるシステム、方法、およびコンピュータプログラム製品を使用して、前述の技術的な問題の1つまたは複数が解決される。特に、本発明の少なくとも1つの実施には、共有するように意図されたディスプレイ画面の領域と、さらに共有しないように意図されたディスプレイ画面の領域とを識別するサーバー ( または送信側コンピュータ ) が含まれる。そして、サーバーは非共有コンテンツを1つまたは複数のシンプルなイメージファイル ( 単純な画像ファイル ) に置き換え、非共有コンテンツの代わりにそのシンプルなイメージファイルを送信する。従って、機密情報を隠すことができ、クライアントコンピュータシステムからの何らかの顕著なプロセスアクション ( p r o c e s s i n g a c t i o n ) を要求する必要はない。

#### 【 0 0 1 1 】

たとえば、本発明の1つまたは複数の実施による共有アプリケーションの描画コマンドと隠しアプリケーションの描画コマンドとをフィルタリングする1つの方法には、少なくとも1つのリモートクライアントにおいて表示されるように意図された1つまたは複数の共有描画コマンドを識別することが含まれる。本方法には、さらにリモート ( 遠隔 ) クライアントから隠されるように意図された1つまたは複数の隠し描画コマンドを識別することも含まれる。さらに、本方法には、1つまたは複数の共有描画コマンドをビットマップにレンダリングすることと、同様に1つまたは複数の隠し描画コマンドの代替イメージをビットマップにレンダリングすることとが含まれる。そういう次第なので、隠さなければならないと意図されたコンテンツの代わりに代替イメージを送信できる。

#### 【 0 0 1 2 】

さらに、リモートクライアントと表示を共有する場合に停止 ( p a u s e ) 機能を実施する方法は、サーバーシステムの1つまたは複数のアプリケーションからの複数の描画コマンドを識別する動作が含まれる。この場合は、複数の描画コマンドの少なくとも1つはクライアントコンピュータシステムで表示されるように意図されている。さらに、本方法には、サーバーの1つまたは複数のアプリケーションのいずれかからの停止命令を受信することと、さらにミラードライバが対応する停止イメージをビットマップにレンダリングすることとが含まれる。このような事情なので、複数の描画コマンドの代わりに停止イメージをクライアントコンピュータシステムに送信できる。

## 【 0 0 1 3 】

「課題を解決するための手段」では、いくつかの概念について、後述の「発明を実施するための最良の形態」で詳細に説明する前に、簡略化した形で紹介している。この項は、特許請求の範囲に記載されている内容の重要な特徴または必須の特徴を示すものではない。また、こうした主題の範囲を判断するための補助として使用するものでもない。

## 【 0 0 1 4 】

本発明のその他の特徴と利点は、この後の説明で示すが、こうした説明からある程度は明らかになるであろうし、あるいは、本発明の実施によって理解できるであろう。本発明の特徴と利点は、添付の特許請求の範囲に詳細に記載する手段と組み合わせによって実現され、達成される。本発明の前述の特徴およびその他の特徴は、以下の説明と添付の特許請求の範囲によってさらに明らかになるであろうし、あるいは、本明細書の以下の部分に示す本発明の実施によっても理解できるものである。

## 【 0 0 1 5 】

本発明の前述の利点と特徴およびその他の利点と特徴を実現する方法を説明するために、以上で簡単に説明した本発明のより詳細な説明をその特定の実施形態に関連付けて記述し、添付の図面に示す。これらの図面が本発明の例示的な実施形態を示すにすぎないこと、したがって本発明の範囲を限定するものとは見なされないことを理解した上で、添付の図面を使用しながら本発明についてより具体的に説明する。

## 【発明を実施するための最良の形態】

## 【 0 0 1 6 】

本発明の実施は、ネットワークを介して接続されたコンピュータシステム間でアプリケーションの表示の共有に関する効率を高めるシステム、方法、およびコンピュータプログラム製品にまで及ぶ。具体的に、本発明の少なくとも1つの実施には、共有するように意図されたディスプレイ画面の領域と共有しないように意図されたディスプレイ画面の領域とを識別するサーバー（または送信側コンピュータ）が含まれる。ここで、サーバーは非共有コンテンツを1つまたは複数のシンプルなイメージファイルに置き換え、非共有コンテンツの代わりにこのシンプルなイメージファイルを送信する。このようにして機密情報を隠すことができ、必ずしもクライアントコンピュータシステムが重要なプロセスアクションを実行する必要はない。

## 【 0 0 1 7 】

本明細書および特許請求の範囲からより詳細に理解されるように、こうしたフィルタの動作は非常にシンプルなりモート表示プロトコルを使用して実行でき、必ずしもサーバー側またはクライアント側から膨大な処理を実行する必要はない。一般に、「リモートビュー（remote view）」、「リモート表示（remote viewing）」、または「リモート共有（remote sharing）」という用語は、遠隔地に配置された別のコンピュータシステムの少なくとも1つのディスプレイ画面データを表示する1つのコンピュータシステムの一般的な概念を示している。また、リモート表示は本明細書で機能について説明されているが、表示および隠し表示に関するかなりきめ細かな情報を受信側コンピュータシステムに送信できるので、必ずしも大きなネットワーク帯域幅を必要とはしない。さらに、こうした共有アプリケーション領域と隠しアプリケーション領域のリモート表示手続きを使用した通信は、リモートデスクトッププロトコル（すなわち「RDP」）などの既存のプロトコルを使用して実行できるので、サーバー側からもクライアント側からもプロトコルの大幅な変更や関連の管理は必ずしも必要ではない。

## 【 0 0 1 8 】

図1Aに示すように、たとえば、送信側コンピュータ100はネットワーク140を経由してアプリケーションデータを受信側コンピュータ150に送信する。たとえば、受信側コンピュータ150（たとえばクライアントコンピュータシステム）は送信側コンピュータ100（たとえばサーバーコンピュータシステム）に接続してアプリケーション110などのさまざまなリソースを共有し、これを受信側コンピュータ150のディスプレイ155にローカルに表示する。あるいは、送信側コンピュータ100のエンドユーザー（

端末使用者)は受信側コンピュータ150のエンドユーザーとディスプレイ画面を共有する。ただし、送信側コンピュータ100のディスプレイ画面105にはアプリケーション110のデータとアプリケーション115のデータが表示されている。いずれにしても、送信側コンピュータ100のディスプレイ画面105に表示できるデータの少なくとも一部は、受信側コンピュータ150のディスプレイ画面155に表示できるように意図されている。さらに、図1Aはアプリケーション110が共有され、アプリケーション115は隠されることを示している。たとえば、アプリケーション115には別の受信側コンピュータシステム(図示せず)からアクセスできる。つまり、送信側コンピュータ100のエンドユーザーが受信側コンピュータ150には表示したくないデータ(たとえば銀行口座が表示されたブラウザなど)は別のコンピュータに表示される。

10

#### 【0019】

アプリケーション110、115がディスプレイ105で目に見える変化となる描画操作を実行すると、ドライバインターフェイス125はこのデータをディスプレイドライバ130に渡す。1つの実施では、ドライバインターフェイス125は、デバイスインターフェイスコンポーネント(たとえば、グラフィックスデバイスインターフェイス("GDI: Graphics Device Interface"))であり、アプリケーションプログラムからのグラフィカルな関数の呼び出しを取得し、対応するグラフィカルな命令を適切なデバイスドライバに供給する。一般に、ドライバインターフェイス125は、アプリケーションプログラムとディスプレイドライバの間を仲介でき、さらに場合によっては以下でより詳細に説明するフィルタ機能性を提供するように設定された任意のコンピュータプログラムコードであることができる。

20

#### 【0020】

たとえば、図1Aはドライバインターフェイス125が共有アプリケーション110から共有表示データ120を受信し、隠しアプリケーション115から隠し表示データ121を受信することを示している。一般に、データ120および121は少なくとも幾何学的な描画データを備えており、こうしたデータを使用して、対応するアプリケーションで生成されたイメージをレンダリングできる。ここで、ドライバインターフェイス125は、ローカル(現地)ディスプレイ用の適切なディスプレイドライバ(この場合はドライバ130)を識別し、識別されたディスプレイドライバにデータ120および121を渡す。これでディスプレイドライバ130はデータをレンダリングでき、これをサーバーのディスプレイ105にローカルに表示するのが一般的である。具体的に、アプリケーションはサーバー100でローカルにホストされ、表示されるため、ドライバインターフェイス125はローカルなフィルタ情報またはフィルタ機能を提供する必要はない。

30

#### 【0021】

また、図1Aはドライバインターフェイス125がミラードライバ135と通信することも示している。ミラードライバは、「リモートビュー」ドライバ(たとえばRDPドライバ)と通信し、アプリケーション110、115から受信した描画コマンドをそのドライバに提供する。一般に、ミラードライバ135は、元は1つのディスプレイデバイスドライバ用であった描画コマンドの複製の描画コマンドを受信でき、こうした複製(または「ミラー(mirrored)」)の形の描画コマンドをレンダリングできる任意のタイプのコンポーネントまたはモジュールであることができる。図示しているように、たとえば、ミラードライバ135はドライバインターフェイス125から描画コマンドを受信する。こうしたコマンドは、同時にディスプレイドライバ130にも送信される。

40

#### 【0022】

しかし、図1Aはドライバインターフェイス125が共有データ120をディスプレイドライバ130に送信するのと同様にミラードライバ135に送信するが、フィルタリング命令123と隠しデータ121もミラードライバ135に送信することも示している。1つの実施では、ドライバインターフェイス125は、描画命令が特定の領域(たとえば共有ディスプレイ画面)に関連する場合にこうした命令のみをミラードライバ135に送信するように設定することもできる。いずれの場合も、フィルタリング命令123は非共

50

有アプリケーション 115 用の幾何学的な命令と、データ 121 は共有されないという指示を備えることができる。または、アプリケーション 115（またはアプリケーション 110、あるいは図示されていない別のコンポーネント）によって選択された任意のイメージファイルを備えることもできる。受信時に、ミラードライバ 135 はデータ 120 および 123 を処理し、対応する描画を送信機側ビットマップ 137a に作成する。

【0023】

しかし、この場合は、ミラードライバ 135 はイメージをデータ 120（すなわち「アプリケーション 110」）に対応する送信機側ビットマップ 137a として描画し、メッセージ 123 に含まれる新しい命令に基づいて、隠しデータ 121 の代わりとなる別の代替イメージ 131 を送信機側ビットマップ 137a として準備する。送信側コンピュータ 100 は、次いでこうしたイメージを適切なプロトコル（たとえば RDP）によるグラフィックスコマンドとして準備し、さらにこうしたグラフィックスコマンドに対応するネットワークパケットを準備する。次に、送信側コンピュータ 100 はネットワーク 140 を経由してこうしたネットワークパケットを送信し、最終的には受信側コンピュータ 150 において対応する受信機側ビットマップ 137b となる。このようなネットワークパケットには、描画コマンドに対応する生のデータに加えて、代替イメージ 131 に付随する命令（たとえば、代替イメージ 131 を並べて表示（tile）か、拡大して表示か、またはそれ以外の何らかの方法で表示かなどについて）を含めることもできる。

【0024】

受信側コンピュータ 150 で、対応するビューアーアプリケーション（図示せず）は受信機側ビットマップ 137b からネットワークパケットのデータストリームを、データ 120 および代替イメージ 131（および対応する表示命令）として読み込み、このデータをドライバインターフェイス 127 に送信する。使用するプロトコル（たとえば RDP など）によって、受信側コンピュータ 150 は代替イメージ 131 をキャッシュに保存して後で利用することも、代替イメージ 131 を代わりに使用する他の特定のイメージに置き換えることもできる。次に、ドライバインターフェイス 127 はデータをディスプレイドライバ 145 に渡し、ディスプレイドライバはデータをレンダリングしてクライアントのディスプレイ画面 155 に表示する。このようにして、クライアントのディスプレイ画面 155 には送信側コンピュータのディスプレイ画面 105 とほとんど同じ、アプリケーション 110 の同一のコピーが表示される。しかしながら、ディスプレイ画面 105 とは異なり、ディスプレイ画面 155 ではアプリケーション 115 は隠されるため、アプリケーション 115 の代わりにフィルタリング命令 123（またはそれ以外の特定の選択された代替イメージ）が表示される。

【0025】

従って、送信側コンピュータ 100 は隠しアプリケーションデータ領域と共有アプリケーションデータ領域を決定するための該当する処理のほとんどすべてを実行し、受信側コンピュータシステム 150 は単に受信したデータを読み込んでレンダリングするだけですむ。これにより、受信側コンピュータシステム 150 は、すでに適切に装備されている機能を実行するだけでよく、受信側コンピュータシステムにインストール済みのリソースに対する追加の更新または拡張を受信する必要はない。さらに、このことにより、本発明の実施は古いバージョンのリモート表示クライアント/サーバーモデルおよび/または関連のプロトコルと容易に互換できる。

【0026】

前述の原理のそれ以外の態様は、データのフィルタを、アプリケーションを介したユーザーモードでなくドライバインターフェイスおよび/または対応するドライバによってカーネルモードで追跡することである。そういう次第なので、さまざまなドライバは本質的に同期およびファンネリング（funneling）エージェントとして動作でき、共有するものはどれか、共有しないものはどれかという直接的な認識に基づいて、イメージをレンダリングして送信することができる。たとえば、アプリケーションが共有とするか非共有とするかを示す指定を受信すると、この指定をアプリケーションの送信データ（た

10

20

30

40

50



例えば、１２０、１２１）に直接関連付けることができ、さらにミラードライバ１３５およびドライバインターフェイス１２５に送信できる。アプリケーションデータが共有されないという指定をミラードライバ１３５が受信したときには、ミラードライバ１３５は送信機側ビットマップ１３７a内にデフォルトのイメージを直ちに描画するか、または送信機側ビットマップ１３７a内で隠しコンテンツを特定のデフォルトのイメージで置き換える。このように、共有および非共有の概念と、受信したデータとの同期はミラードライバ１３５で実現できるので、ミラードライバ１３５はアプリケーションから同期情報を受信する場合のいく分かの遅延によって非共有データを送信する状況を回避できる。

【００２７】

図１Ｂは、リモート表示環境またはリモート共有環境で前述の技術を実施できる別の方法を示している。具体的に、図１Ｂはブロードキャストコンピュータシステム（たとえば、学習環境で送信側コンピュータ１００を使用するインストラクタ）がシンプルなりモート共有プロトコル（たとえばRDP）を使用して表示（presentation）を停止できる方法を示している。停止コマンドを実施すると、すべてのデータが１つまたは複数の対応する任意の受信側コンピュータシステム（たとえば受信側コンピュータ１５０）に送信される前に、代替イメージによって隠すことができる。

【００２８】

たとえば、図１Ｂは図１Ａと同様にドライバインターフェイス１２５が共有データ１２０と隠しデータ１２１を受信することを示している。しかし、図示されたケースでは、ドライバインターフェイス１２５はたとえば共有アプリケーション１１０からコラボレーション（collaboration）APIを使用してさらに停止命令１６０も受信する。ドライバインターフェイス１２５は、ここで前と同様にデータ１２０および１２１をディスプレイドライバ１３０に渡し、さらにデータ１２０および１２１に加えて停止命令１６０をドライバインターフェイス１２５からミラードライバ１３５に渡す。停止命令１６０を使用する場合は、たとえば、送信ユーザー（インストラクタなど）は隠そうとしているアプリケーションの前面の表示に使用するアプリケーションを変更してもよい。インストラクタは、隠しアプリケーションが新しいウィンドウまたはアプリケーションを閉じる時、または開くときに瞬間的に表示されることがないように、受信側コンピュータシステムのディスプレイ画面全体を瞬間的な停止イメージで隠したい場合がある。それに応じて、インストラクタは対応するアプリケーションに停止命令１６０を送信するように要求する。

【００２９】

ドライバインターフェイス１２５は、ミラードライバ１３５に、アプリケーション１１０、１１５から送信されたアプリケーションデータを渡し、さらに停止命令１６０を渡す。命令１６０は、対応するアプリケーション（たとえば１１０および／または１１５）が現在の表示の境界（borders）に関して識別した情報と、これに続くこうした境界の間の少なくとも一部を適切に選択された停止イメージで置き換えるという命令の準備に基づくことができる。ミラードライバ１３５は、データ１２０および１２１を受信すると同時に停止イメージ１６１に置き換えることができる。

【００３０】

いずれの場合も、ミラードライバは適切な停止イメージを送信機側ビットマップ１３７aに描画し、データ１２０および１２１の一部またはすべてを、停止イメージ１６１に置き換える。図１Ａで説明したように、ここで送信側コンピュータ１００はネットワーク１４０を経由してこのデータをネットワークパケットの形で受信機側ビットマップ１３７bに送信する。受信側コンピュータ１５０は、次いで必要に応じてこのデータを読み込み、中継し、さらにレンダリングし、最終的に停止イメージ１６１をディスプレイ１５５に表示する。さらに、受信側コンピュータ１５０は停止イメージ１６１をキャッシュに保存して後で利用できる。また、必要に応じて停止イメージ１６１を適切に選択された別の停止イメージで置き換えることもできる。しかし、レンダリングされ、表示されるにもかかわらず、受信側コンピュータ１５０は送信側コンピュータ１００が特に隠されるようにまた

10

20

30

40

50

は表示されないように意図されたコンテンツを少なくとも部分的には表示できない。まず、最初に、こうしたコンテンツは送信側コンピュータ 100 から受信されないためである。

#### 【0031】

たとえば、概略図 1 A ~ 図 1 B と対応する説明文には、デスクトップ共有環境で共有データの概念と隠し（あるいは非共有または非表示）データの概念を実施できる多くのコンポーネントおよび手段が提供されている。本発明の実施については、方法を実現する一連の動作の観点から説明することもできる。たとえば、図 2 と図 3 はリモート表示環境で共有アプリケーションの描画コマンドと隠しアプリケーションの描画コマンドとをフィルタリングする方法、および停止機能をリモートで実施する方法のそれぞれについて、その一連の動作を表す流れ図を示している。図 2 と図 3 については、図 1 A ~ 図 1 B のコンポーネントおよびモジュールを参照しながら以下で詳細に説明する。

10

#### 【0032】

具体的に、図 2 は送信側コンピュータで共有アプリケーションの描画コマンドと隠しアプリケーションの描画コマンドとをフィルタリングする方法は、共有描画コマンドを識別する動作 200 を備えることを示している。動作 200 には、リモートクライアントで表示されるように意図された 1 つまたは複数の共有描画コマンドを識別する動作が含まれる。たとえば、ドライバインターフェイス 125 は、共有アプリケーション 110 から描画コマンド 120 を受信する。描画コマンド 120 には、描画コマンドによって定義されたイメージが共有されること、すなわち受信側コンピュータ 150 で表示できることを示す命令が関連付けられている。さらに、図 2 は本方法がさらに隠し描画コマンドを識別する動作 210 を備えることも示している。動作 210 には、リモートクライアントから隠されるように意図された 1 つまたは複数の隠し描画コマンドを識別する動作が含まれる。たとえば、ミラードライバ 135 は（インターフェイス 125 を経由して、またはアプリケーション 115 から直接）データ 121 を受信する。このデータは、受信側コンピュータ 150 で表示されないように意図された描画コマンドを表している。

20

#### 【0033】

さらに、図 2 は本方法がさらに共有描画コマンドをレンダリングする動作 220 を備えることも示している。動作 220 には、1 つまたは複数の共有描画コマンドをビットマップにレンダリングする動作が含まれる。たとえば、ミラードライバ 135 はデータ 120 をアプリケーション 110 から直接またはドライバインターフェイス 125 を経由して受信する。ミラードライバ 135 はこれが共有コンテンツであることを認識するので、ミラードライバ 135 はこのデータを送信機側ビットマップ 137 a（すなわちアプリケーション 110 のイメージ）に描画し、ネットワーク 140 を経由してパケット形式で送信されるように準備する。

30

#### 【0034】

さらに、本方法は代替イメージをレンダリングする動作 230 も含む。動作 230 には、隠されるように意図されたコンテンツの代わりに代替イメージを送信できるように、1 つまたは複数の隠し描画コマンドの代替イメージをビットマップにレンダリングする動作が含まれる。たとえば、ミラードライバ 135 がデータメッセージ 120 および 121 とフィルタリング命令 123 を受信する。この命令は、ディスプレイで隠し境界（boundaries）に関する幾何学的なデータと、メッセージ 120 および 121 に含まれるデータは表示されるように意図されていないことを示している。代わりに、フィルタリング命令 123 にはデータ 120 および 121 の代わりに実際に使用される代替イメージが含まれる。したがって、ミラードライバ 135 は代替イメージ 131 を送信機側ビットマップ 137 a に描画する。

40

#### 【0035】

図 3 は、本発明の実施による類似ではあるが別の方法を示している。この方法には、停止機能を実施する場合に共有アプリケーションの描画情報および / または隠しアプリケーションの描画情報をフィルタリングする動作が含まれる。たとえば、図 3 は本方法が 1 つ

50

または複数のサーバーアプリケーションによる複数の描画コマンドを識別する動作 300 を備えることを示している。動作 300 には、1つまたは複数のサーバーアプリケーションによる複数の描画コマンドの少なくとも1つはクライアントコンピュータシステムでも表示されるように意図されている場合に、こうした複数の描画コマンドを識別する動作が含まれている。たとえば、図 1B に示すように、ミラードライバ 135 は、たとえばドライバインターフェイス 125 を経由して、アプリケーション 110、115 などからデータ 120、121 のメッセージを受信する。この情報は、たとえば、送信側コンピュータ 100 からネットワーク 140 を経由して1つまたは複数の他の受信側コンピュータシステムに送信された表示の間に受信できる。

【0036】

さらに、図 3 は本方法がさらに停止命令を受信する動作 310 を備えることを示している。動作 310 には、サーバーでホストされる1つまたは複数のアプリケーションから停止命令を受信する動作が含まれる。たとえば、サーバー 100 のドライバインターフェイス 125 は停止命令 160 を受信し、これを直接ミラードライバ 135 に渡す。あるいは、ミラードライバ 135 は停止命令 160 をアプリケーション 110 または 115 から直接受信する。さらに、図 3 は本方法が任意の描画コマンドの代わりに停止イメージをレンダリングする動作 320 を備えることを示している。動作 320 には、停止イメージが任意の複数の描画コマンドの代わりにクライアントコンピュータシステムに送信されるように、停止イメージをビットマップにレンダリングする動作が含まれる。たとえば、図 1B はドライバインターフェイス 125 が停止イメージ 161 (または対応する同等の命令) をミラードライバ 135 に送信することを示している。次に、ミラードライバ 135 は、データ 160 がパケット化され、ネットワーク 140 を経由して受信側コンピュータ 150 に送信される前に、データ 160 を送信機側ビットマップ 137a に描画する。

【0037】

したがって、本発明の実施は、共有アプリケーションまたは非共有アプリケーションから送信されたデータに対応する共有データおよび非共有データ(または隠しデータ)を効率的にレンダリングするための多くのコンポーネントおよび方法を提供する。特に、本発明の実施は、実行する必要がある多くの処理が一般に他の方法に比べてシンプルであるため、多くの技術的な利点を提供する。さらに、より複雑な処理の側面は送信側コンピュータでのみ実行すればよい。このように、本発明の実施はすでにリモート共有プロトコル(たとえば RDP)が組み込まれているシステムに容易に適用でき、非常に多くの共有表示情報/非共有表示情報を提供でき、既存のリソースおよびプロトコルの更新または拡張は必ずしも必要ではない。

【0038】

本発明の実施形態は、以下で詳説するように、さまざまなコンピュータハードウェアを含む専用コンピュータまたは汎用コンピュータを備えていてもよい。特に、本発明の範囲に含まれる実施形態には、コンピュータ実行可能命令またはデータ構造を格納して運搬したり所持したりするための、コンピュータ可読媒体を含めることもできる。こうしたコンピュータ可読媒体は、汎用コンピュータまたは専用コンピュータからアクセスできる任意の媒体でよい。例として、こうしたコンピュータ可読媒体には、RAM、ROM、EEPROM、CD-ROM などの光ディスクストレージ、磁気ディスクストレージなどの磁気ストレージデバイス、あるいは必要なプログラムコード手段を汎用コンピュータまたは専用コンピュータからアクセスできるコンピュータ実行可能命令の形またはデータ構造の形で運搬し、格納するために使用できる他の任意の媒体を含めることができるが、これらに限定はされない。

【0039】

情報がネットワークまたは別の通信接続(有線、無線、有線と無線の組み合わせのいずれかを問わない)を経由してコンピュータに転送または提供されると、コンピュータはこうした接続をコンピュータ可読媒体として適切に表示する。したがって、こうした任意の接続をコンピュータ可読媒体と呼ぶのは適切である。上記の組み合わせも、コンピュータ

10

20

30

40

50

可読媒体の範囲内に含まれるものとする。

【 0 0 4 0 】

コンピュータ実行可能命令は、たとえば汎用コンピュータ、専用コンピュータ、または専用の処理デバイスが特定の機能または機能グループを実行するための命令およびデータを含む。本主題について、構造上の特徴および／または方法論的な動作に固有のことばで説明してきたが、添付の特許請求の範囲によって定義される本主題は必ずしも前述の特定の機能および動作には限定されないことを理解されたい。むしろ、前述の特定の機能および動作は、特許請求の範囲を実施する例示的な形態として開示されている。

【 0 0 4 1 】

本発明は、その精神または本質的な特性を逸脱することなく、他の特定の形態でも具体化できる。以上に説明されている実施形態は、あらゆる態様において例示的なものにすぎず、限定的なものではない。したがって、本発明の範囲は、前述の説明ではなく、添付の特許請求の範囲によって指定される。特許請求の範囲と同等の意味および範囲を逸脱しないすべての変更は、本発明の特許請求の範囲に含まれるものとする。

【図面の簡単な説明】

【 0 0 4 2 】

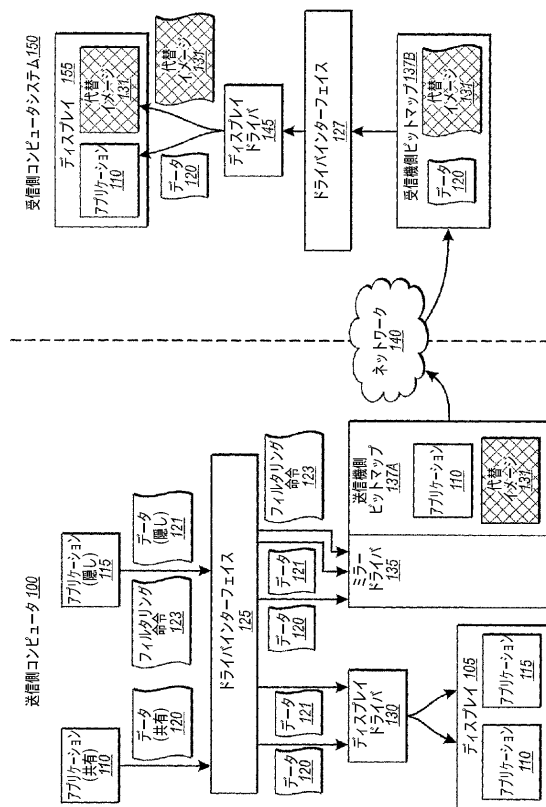
【図 1 A】本発明の実施によるアプリケーション共有環境において、ドライバインターフェイスが共有アプリケーションからの描画コマンドと隠しアプリケーションからの描画コマンドとをフィルタリングすることを示す概略図である。

【図 1 B】図 1 A に示す概略図においてドライバインターフェイスがアプリケーションから受信した停止命令に応答することを示す概略図である。

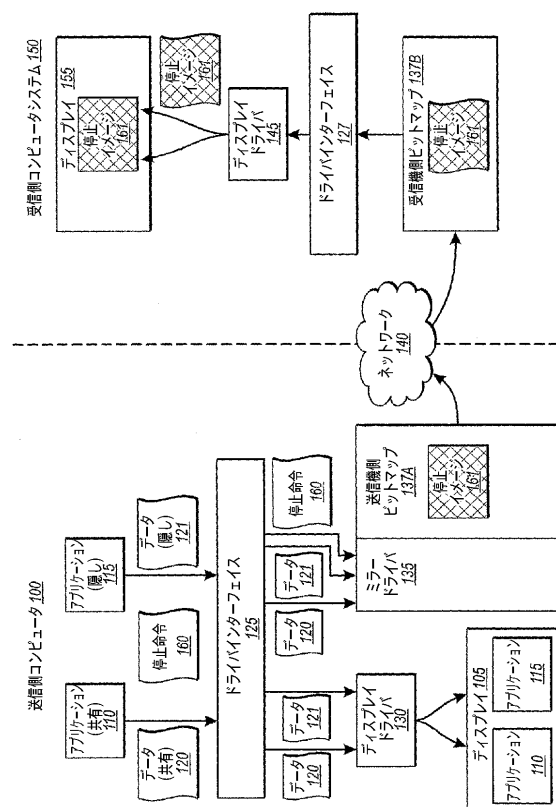
【図 2】本発明の実施によるアプリケーション共有環境において共有描画コマンドと隠し描画コマンドとをフィルタリングする方法の一連の動作を示す流れ図である。

【図 3】本発明の実施によるアプリケーション共有環境において停止機能を実施する方法の一連の動作を示す流れ図である。

【図 1 A】



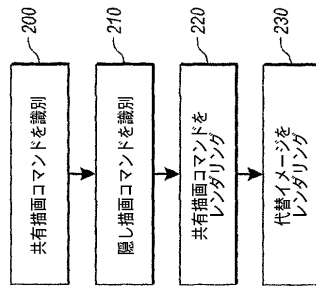
【図 1 B】



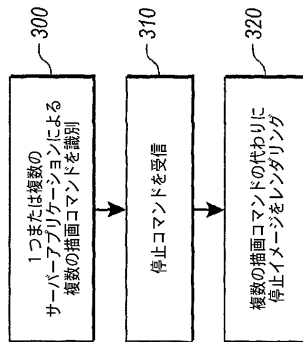
10

20

【図 2】



【図 3】



---

フロントページの続き

(72)発明者 エルトン ソール

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 和田 財太

(56)参考文献 特開2004-185568(JP,A)

特開2005-018220(JP,A)

特開平11-065975(JP,A)

特開2003-085135(JP,A)

(58)調査した分野(Int.Cl., DB名)

G06F 21/24

G06F 3/14

G06T 1/00