

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第5455930号
(P5455930)

(45) 発行日 平成26年3月26日 (2014. 3. 26)

(24) 登録日 平成26年1月17日 (2014. 1. 17)

(51) Int. Cl. F I
G06T 11/00 (2006.01) G O 6 T 11/00 1 0 0 A
G06F 13/00 (2006.01) G O 6 F 13/00 5 4 0 A

請求項の数 21 (全 19 頁)

(21) 出願番号	特願2010-545038 (P2010-545038)	(73) 特許権者	500046438
(86) (22) 出願日	平成21年1月6日 (2009. 1. 6)		マイクロソフト コーポレーション
(65) 公表番号	特表2011-511367 (P2011-511367A)		アメリカ合衆国 ワシントン州 9805
(43) 公表日	平成23年4月7日 (2011. 4. 7)		2-6399 レッドモンド ワン マイ
(86) 国際出願番号	PCT/US2009/030201		クロソフト ウェイ
(87) 国際公開番号	W02009/099689	(74) 代理人	100140109
(87) 国際公開日	平成21年8月13日 (2009. 8. 13)		弁理士 小野 新次郎
審査請求日	平成23年11月25日 (2011. 11. 25)	(74) 代理人	100089705
(31) 優先権主張番号	12/024, 848		弁理士 社本 一夫
(32) 優先日	平成20年2月1日 (2008. 2. 1)	(74) 代理人	100075270
(33) 優先権主張国	米国 (US)		弁理士 小林 泰
		(74) 代理人	100080137
			弁理士 千葉 昭男
		(74) 代理人	100096013
			弁理士 富田 博行

最終頁に続く

(54) 【発明の名称】 グラフィック遠隔化アーキテクチャ

(57) 【特許請求の範囲】

【請求項 1】

メモリーと、

前記メモリーと作動可能に接続されるプロセッサと、

1つ以上のクライアントに対応している前記メモリー中のユーザー空間であって、前記ユーザー空間が、少なくとも1つの三次元(3D)コマンド及び少なくとも1つの3Dオブジェクトを含んでいるデータトラフィックを監視するための抽象化プログラムモジュールを含んでおり、前記抽象化モジュールが更に、前記1つ以上のクライアントの内の第1のクライアントの1つ以上の処理能力と、少なくとも1つの前記3Dコマンド及び少なくとも1つの前記3Dオブジェクトに関連する3Dアプリケーションのタイプに基づいてフ

10

レーム転送レートを制限し、累進的なテクスチャレンダリング方法を用いて帯域幅を前記第1のクライアントに割り当てることによって、前記第1のクライアントによる帯域幅消費を規制するように構成されているものと、

を含み、

前記メモリー及びプロセッサが、

前記ユーザー空間に関連する装置の切断情報を受け取り、当該切断情報を受け取った時に前記ユーザー空間に関連するクライアント装置の切断又は再接続処理を起動するように構成されているサーバーコンピューター。

【請求項 2】

前記メモリーが、前記3Dコマンド、前記3Dオブジェクト及びデータストリームをス

20

トアしているプログラムデータを含むことを特徴とする請求項 1 記載のサーバーコンピューター。

【請求項 3】

前記ユーザー空間が、クライアント装置における、ディスプレイアダプターの利用可能性及び機能に関する問い合わせをするリモートアダプターを含むことを特徴とする請求項 1 記載のサーバーコンピューター。

【請求項 4】

前記リモートアダプターが、前記抽象化プログラムモジュールと前記クライアント装置のプラグインとの間に通信チャネルを確立し、前記プラグインが、コマンド及び 3 D オブジェクトを受信することと、を特徴とする請求項 3 記載のサーバーコンピューター。

10

【請求項 5】

前記抽象化プログラムモジュールが、3 D アプリケーション中で 3 D 画像をレンダリングする機能を含むアプリケーション・プログラム・インターフェース (API) を介し、前記 3 D アプリケーションから前記 3 D コマンド及び 3 D オブジェクトを受信するように構成されることを特徴とする請求項 1 記載のサーバーコンピューター。

【請求項 6】

前記抽象化プログラムモジュールが、前記 3 D オブジェクトに識別子を割り当てることを特徴とする請求項 1 記載のサーバーコンピューター。

【請求項 7】

前記抽象化プログラムモジュールが、前記 3 D コマンド及び 3 D オブジェクトをデータストリームの中に抽象化し、前記データストリームを前記 1 つ以上のクライアントに送信することを特徴とする請求項 1 記載のサーバーコンピューター。

20

【請求項 8】

前記抽象化プログラムモジュールが、圧縮及びデシメーションによる集中型符号化を実行することを特徴とする請求項 1 記載のサーバーコンピューター。

【請求項 9】

メモリーと、

前記メモリーと作動可能に接続されるプロセッサと、

3 D コマンド及び 3 D オブジェクトをインクルードしたデータストリームを受信し、抽出する前記メモリー中の 3 D プラグインであって、前記 3 D コマンドはグラフィックレンダリングセッションにおいて前記 3 D オブジェクトを用い、前記 3 D プラグインの機能が、前記サーバーコンピューターとの対話用のフレームレートを決定するためにサーバーコンピューターに転送されるものと、

30

前記抽出した 3 D コマンドをレンダリングするグラフィック・プロセッシング・ユニット (GPU) と、

クライアントコンピューターがサーバーコンピューターとの接続を喪失した時に、前記サーバーコンピューターに通知する、装置切断通知ユニットであって、前記クライアントコンピューターとの切断時に、前記 3 D プラグインが再接続までレンダリングセッションを維持するように構成されるものと、

を含むクライアントコンピューター。

40

【請求項 10】

前記 3 D プラグインが、前記データストリームを解析し、3 D コマンドを抽出することを特徴とする請求項 9 記載のクライアントコンピューター。

【請求項 11】

前記 3 D プラグインが、前記データストリーム中の前記 3 D オブジェクトに関連付けられたハンドルを識別することによって 3 D オブジェクトを生成することを特徴とする請求項 9 記載のクライアントコンピューター。

【請求項 12】

前記ハンドルが、グラフィックアプリケーションに関連付けられたリソースのアドレスを含んでいるエントリーを含むことを特徴とする請求項 11 記載のクライアントコンピュ

50

ーター。

【請求項 13】

前記 3D プラグインが、3D グラフィックをフルスクリーンのユーザーインターフェースに提示することを特徴とする請求項 9 記載のクライアントコンピューター。

【請求項 14】

前記 3D プラグインが、同期化情報をサーバーに送信することを特徴とする請求項 9 記載のクライアントコンピューター。

【請求項 15】

前記 3D プラグインが、クライアントサイド専用のレンダリングプロトコルに基づいて出力グラフィックをレンダリングすることを特徴とする請求項 9 記載のクライアントコンピューター。

10

【請求項 16】

前記 GPU が、前記データストリームを送信するサーバーコンピューターのウィンドウに生じる幾何学的変化を生成するためのプロキシウィンドウを生成することを特徴とする請求項 9 記載のクライアントコンピューター。

【請求項 17】

プロセッサで実行された時、計算機に方法を実行させる計算機可読命令を記憶する計算機記憶媒体であって、前記方法が、

1 つ以上のグラフィックアプリケーションコマンドを受信し、識別するステップと、

前記 1 つ以上のグラフィックアプリケーションコマンドをグラフィックドライバーに伝達するステップと、

20

前記 1 つ以上のアプリケーションコマンド及び前記グラフィックドライバーによって生成されたオブジェクトを、前記 1 つ以上のアプリケーションコマンドに基づいてデータストリームに抽象化するステップと、

クライアントコンピューターから装置切断情報が受け取られた時に、コンピュータ上のサーバー状態とクライアントコンピューター上のクライアント状態とを再同期させるための装置切断復旧処理を起動するステップと、

前記クライアントコンピューターによる帯域幅消費を、前記アプリケーションコマンド及び前記オブジェクトに関連するアプリケーションのタイプと、前記クライアントデバイスの 1 つ以上の処理能力とに基づいてフレームレートを制限することによって規制するス

30

テップと、
を含む計算機記憶媒体。

【請求項 18】

伝達するステップが、アプリケーションプログラムインターフェースを介することを特徴とする請求項 17 記載の計算機記憶媒体。

【請求項 19】

前記抽象化するステップが、生成されたコマンド及びオブジェクトを隠されたデータとして抽象化するステップを含むことを特徴とする請求項 17 記載の計算機記憶媒体。

【請求項 20】

前記方法が更に、前記データストリームをネットワーク上の 1 つ以上のクライアントに送信するステップを含む請求項 17 記載の計算機記憶媒体。

40

【請求項 21】

前記方法が更に、前記データストリームを圧縮するステップを含む請求項 20 記載の計算機記憶媒体。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、計算システムにおけるグラフィック処理に関し、具体的には、リモート計算環境におけるグラフィック処理に関する。

【背景技術】

50

【0002】

[0001] フォト編集、ゲームなどのように高性能グラフィックアプリケーションを生成するためのグラフィックソフトは、複雑な計算を利用する動画及びその他のツールを含み得る。通常、そのような高性能グラフィックアプリケーションは、複雑な計算を管理するためのグラフィックハードウェアの支援を提供する様々な技術を使用する。例えば、Microsoft（登録商標）Windows（登録商標）オペレーティングシステム上で実行されるグラフィックアプリケーションは、グラフィックハードウェアの支援を提供するDirect3Dレンダリング技術を使用し得る。Direct3Dは、グラフィックアプリケーションの中で3D画像をレンダリングするためのコマンドを含んでいる三次元（3D）グラフィック・アプリケーション・プログラミング・インターフェース（API）である。

10

【0003】

[0002] 今やほとんどのパーソナルコンピュータ（PC）は、そのようなグラフィックハードウェアの支援を提供しているが、そのような技術を遠隔化する効率的な方法はどこにも存在しない。例えば、リモートセッションにおいて、グラフィックアプリケーションは、ウィンドウサイズもユーザー接続数も率に応じ増やすことができず、安定性問題ももたらし得る。更に、グラフィックス・アクセラレーションは、ビデオ・グラフィックス・アダプター（VGA）を実行するサーバーコンピュータにおいても仮想計算機においても利用され得ない。結果として、リモートセッションを介し実行するグラフィックアプリケーションは、質の悪いエンドユーザー体験をもたらし得る。

20

【発明の概要】

【発明が解決しようとする課題】

【0004】

本発明の目的は、リモートクライアントにおいて、3Dグラフィック画像のレンダリングを可能にするDirect3D遠隔化アーキテクチャを用いた自己記述型データレンダリングを実装する技法を提供することである。

【課題を解決するための手段】

【0005】

[0003] この概要は、以下の「発明を実施するための形態」において更に記載されているグラフィック遠隔化アーキテクチャに関連する簡易化した概念を紹介するために提供されている。この概要は、請求対象項目の本質的な特徴を特定することも意図されておらず、請求対象項目の範囲を決定する際に使用することも意図されていない。

30

【0006】

[0004] 実施形態において、サーバーコンピュータ上にホスティングされる三次元（3D）アプリケーションは、リモートクライアントコンピュータによって使用される。3Dアプリケーションに関連付けられたグラフィックは、リモートサーバーにおいてデータストリームの中に抽象化され、クライアントコンピュータに送信されるグラフィックに対応している3Dコマンド及び3Dオブジェクトの使用によってリモートクライアントコンピュータにおいて生成され、レンダリングされる。3Dコマンド及び3Dオブジェクトがデータストリームから抽出され、実行され、クライアントコンピュータにグラフィック画像を生成する。

40

【0007】

[0005] 詳細な説明を添付の図を参照し説明する。図面において参照番号の最上位の桁は、参照番号が最初に出現する図面を識別している。異なる図面における同一の参照番号の使用は、同等又は同一の項目を示している。

【図面の簡単な説明】

【0008】

【図1】 [0006] 3D遠隔化アーキテクチャを実装するための例示的ネットワーク環境を示すブロック図である。

【図2】 [0007] 3D画像のリモートレンダリングを実装するための例示的計算サーバーを

50

示すブロック図である。

【図3】[0008] 3D画像のリモートレンダリングを実装するための例示的クライアントを示すブロック図である。

【図4】[0009] 3D遠隔化アーキテクチャの例示的実装を示すブロック図である。

【図5】[0010] 3Dグラフィックアプリケーション中のコマンドを抽象化するための例示的プロセスを示す流れ図である。

【図6】[0011] 3D画像をレンダリングするための例示的プロセスを示す流れ図である。

【発明を実施するための形態】

【0009】

[0012]この開示は、リモートに位置するクライアントにおいて三次元又は3D画像のレンダリングを容易にし、リモートセッションにおいて、改善されたエンドユーザー体験を達成するための遠隔化アーキテクチャを実装する方法及びシステムに関連する。通常、Terminal Service（登録商標）環境のようなリモート環境において、Direct3D又はD3Dコマンドのような任意のグラフィックに集中したアプリケーションコマンドは、サーバーで実行される。コマンド実行後、グラフィック画像などの出力は、サーバーのグラフィックス・プロセッシング・ユニット（GPU）の支援を用いてレンダリングされる。レンダリングされたグラフィック画像は、その後、ビットマップ画像としてクライアント装置に送信され、クライアントサイドにおいて、表示装置を介し表示される。そのような方法にて実行するDirect3D又は三次元アプリケーションは、サーバーの利用可能性、拡張性、サーバー安定性などアーキテクチャ上の要因によって制限され、質の悪いエンドユーザー体験をもたらす。具体的には、圧縮及びその他の特殊方法を使用しているとき、アーキテクチャ上の欠点は、フレームレート制御の欠如及び斬新的なテクスチャレンダリングの不適当性を含み得る。

【0010】

[0013]リモートクライアントにおいて3Dグラフィック画像のレンダリングを可能にするためのDirect3D遠隔化アーキテクチャを実装する技法が、以下の説明に開示される。一実装において、サーバーコンピュータにおいてグラフィック画像をレンダリングして、その後、最終的なビットマップ画像をクライアント装置に送信する代わりに、D3Dコマンドが、その後D3Dコマンドを実行し得るクライアント装置に直接に送信され得る。クライアント端末装置のGPUが、グラフィック画像などの出力をレンダリングし、クライアント端末に画像を表示する。かくしてグラフィック画像は、サーバー端末のGPUを使用せずに、クライアント装置においてレンダリングされ得る。更に、フレームレート制御及び斬新的なテクスチャレンダリング方法を実装し、前述したアーキテクチャ上の欠点を克服するための技法が開示される。

【0011】

[0014]クライアント装置において、3Dグラフィック画像をレンダリングするためのDirect3D遠隔化アーキテクチャを実装する方法及びシステムが、以下の説明に開示される。

【実施例】

【0012】

例示的ネットワーク環境

[0015]図1は、本明細書に記載した技法を実装し、本明細書に記載したエレメントの全体又は一部を代表し得る3D又はDirect3D（D3D）遠隔化アーキテクチャを実装するための例示的ネットワーク又は計算環境（100）を示している。計算環境（100）は、計算環境の例であって、使用されている計算機及びネットワークアーキテクチャの使用範囲及び/又は機能性に対し、どのような制限も示すように意図されていない。

【0013】

[0016]計算環境（100）は、（クライアント（102）としてまとめて参照される）複数のクライアント装置（102-1, 102-2, 102-3, ..., 102-N）を含む。クライアント（102）は、ネットワーク（104）を介し、サーバーコンピュータ

10

20

30

40

50

ー又はサーバー(106)と通信可能に関連付けられる。クライアント(102)は、例えば、デスクトップPC、ノートブックコンピューター、ポータブルコンピューター、個人情報端末(PDA)、ワークステーション、携帯計算装置、又はインターネット接続専用端末など様々な従来の任意の計算装置であり得る。クライアント(102)それぞれは、レンダリング用の異なるプログラムモジュールをストアするメモリーを含む。例えば、クライアント(102-1)は、D3Dプラグイン(112)及びレンダリングを実行するための別のモジュールのような異なるプログラムモジュール(110)をストアし得るメモリー(108)を含む。

【0014】

[0017]ネットワーク(104)は、無線ネットワーク若しくは有線ネットワーク又はその組み合わせであり得る。ネットワーク(104)は、お互いに相互接続された個別のネットワークの集まり及び単一の大規模ネットワーク(例えばインターネット又はイントラネット)としての機能性としても代表され得る。そのような個別のネットワークの例は、ローカルエリアネットワーク(LAN)、広域ネットワーク(WAN)、及びメトロポリタンエリアネットワーク(MAN)を含むがこれらに限定しない。更に、個別のネットワークは、無線ネットワーク若しくは有線ネットワーク又はその組み合わせでもあり得る。

【0015】

[0018]サーバー(106)は、例えば、汎用計算装置、又はサーバークラスター及びメインフレームクラスターであり得る。サーバー(106)は、クライアント(102-1)などのクライアントに対応しているユーザー空間(116)を有するメモリー(114)を含む。ユーザー空間(116)は更に、D3D抽象化モジュール(118)及びリモートアダプター(120)を含む。サーバー(106)のリモートアダプター(120)は、クライアントとの接続を確立し、D3D抽象化モジュール(118)によって抽象化されたレンダリング用グラフィックコマンドを送信する。

【0016】

[0019]一実装において、D3D抽象化モジュール(118)が、ターミナル・サーバー(TS)/D3D抽象化レイヤ上で実行する。D3D抽象化レイヤは、サーバー(106)の物理ハードウェアとD3Dアプリケーションのようなグラフィックアプリケーションとの間に実装される。抽象化レイヤは、ハードウェアの実装詳細をそのようなアプリケーションから隠す。

【0017】

[0020]D3D抽象化モジュール(118)は、クライアント(102-1)などのクライアントが、リモートセッション中、サーバー(106)のD3Dアプリケーションを使用するとき、D3Dアプリケーションから1つ以上のD3Dコマンド及びD3Dオブジェクトを受信する。D3D抽象化モジュール(118)は更に、D3DアプリケーションとD3Dアプリケーションを実行するグラフィックハードウェアとの間の通信を抽象化する。

【0018】

[0021]D3D抽象化モジュール(118)は、ネットワーク(104)上に送信され得るストリーミングデータの中にD3Dコマンド及びD3Dオブジェクトを抽象化する。D3D抽象化モジュール(118)は、かくして、サーバー(106)が、データストリームとして抽象化されたコマンド及びオブジェクトを1つ以上のクライアント(102)に送信可能にする。データストリームは、ネットワーク(104)を介し、リモート・デスクトップ・プロトコル(RDP)、D3D遠隔化プロトコルなどのような異なる遠隔化プロトコルに基づいて送信される。

【0019】

[0022]データストリームをサーバー(106)からクライアント(102-1)に送信するために、リモートアダプター(120)が、ネットワーク(104)上のクライアント(102-1)のD3Dプラグイン(112)と通信チャネルを確立する。リモートアダプター(120)は、D3Dプラグイン(112)からクライアント(102-1)の

10

20

30

40

50

ディスプレイアダプターの利用可能性及び機能をチェックする。D3Dプラグイン(112)からの確認を受信すると、リモートアダプター(120)は、D3D抽象化モジュール(118)とD3Dプラグイン(112)との間に通信チャネルを確立する。

【0020】

[0023] D3D抽象化モジュール(118)は、クライアント(102-1)においてD3Dアプリケーションに関連付けられた抽象化D3Dコマンド及びD3DオブジェクトをD3Dプラグインに送信する。一実装において、D3D抽象化モジュール(118)は、抽象化されたD3Dコマンド及びD3DオブジェクトをD3Dプラグイン(112)に隠されたデータとして送信する。

【0021】

[0024] クライアント端末(すなわちクライアント(102))において、データストリームが、D3Dプラグイン(112)のような1つ以上のD3Dプラグインによって受信され、処理される。D3Dプラグイン(112)は、データストリームを受信し、データストリームを解析し、D3Dコマンドを抽出する。更に、D3Dプラグインは、実際のD3Dオブジェクトを生成し、D3Dコマンドを実行し、クライアント端末におけるGPUを使用しグラフィック画像などの出力をレンダリングする。

例示的サーバー

[0025] 図2は、3D画像リモートレンダリングを実装するための例示的サーバー(106)である。サーバー(106)は、プロセッサ(202)、ネットワークインターフェース(単数又は複数)(204)、入力/出力インターフェース(単数又は複数)(206)、及びメモリ(114)を含む計算装置であり得る。ネットワークインターフェース(単数又は複数)(204)は、クライアント(102)又はサーバー(106)などの多くの計算装置と相互接続するための1つ以上のポートを含み得る。ネットワークインターフェース(単数又は複数)(204)は、有線ネットワーク(例えばLANネットワーク、ケーブルネットワークなど)並びに無線ネットワーク(例えばWLANネットワーク、セルラーネットワーク、衛星ネットワークなど)を含む様々なネットワーク及びプロトコルタイプを用いた通信を容易にし得る。

【0022】

[0026] 入力/出力インターフェース(単数又は複数)(206)は、例えば、スキャナーポート、マウスポート、キーボードポートなどを含んでいて、データをクライアント(102)から受信するか又はそれに送信する。一実装において、サーバー(106)の入力/出力インターフェース(単数又は複数)(206)は、ネットワーク(104)上にD3Dコマンド及びD3Dオブジェクトなどのデータを送信する。サーバー(106)の入力/出力インターフェース(単数又は複数)(206)は、同期化のために、レンダリングされたグラフィック画像を代表とするデータ又はコンテンツなどのデータを1つ以上のクライアント(102)から受信する。

【0023】

[0027] メモリ(114)は、ランダムアクセスメモリ(RAM)のような揮発性メモリ、及び/又は読み出し専用メモリ(ROM)、及びフラッシュRAMのような不揮発性メモリ双方の形式の任意の計算機可読媒体であり得る。メモリ(114)は、通常、プロセッサ(202)によってそれに即座にアクセス可能なデータ及び/又はその上で間もなく作動されるプログラムモジュールを含む。

【0024】

[0028] 一実装において、メモリ(114)は、クライアント(102)それぞれに対応しているユーザー空間(116)及びプログラムデータ(208)を含む。ユーザー空間(116)それぞれは、その他のモジュール(210)に加えてD3D抽象化モジュール(118)及びリモートアダプター(120)を含む。プログラムデータ(208)は、D3Dコマンド(212)、D3Dオブジェクト(214)、データストリーム(216)、及びその他のプログラムデータ(218)を含むパラメータをストアし得る。

【0025】

10

20

30

40

50

[0029] D 3 D 抽象化モジュール (1 1 8) は、D 3 D コマンド (2 1 2) 及び D 3 D オブジェクト (2 1 4) を含んでいるデータトラフィックを監視するように構成され得る。以下、更に詳細に説明されるように、D 3 D 抽象化モジュール (1 1 8) は、D 3 D アプリケーションから D 3 D API を介し D 3 D コマンド (2 1 2) 及び D 3 D オブジェクト (2 1 4) を受信するように構成され得る。D 3 D API は、D 3 D アプリケーション中の 3 D 画像をレンダリングするために使用され得る機能を含む。別の実装において、D 3 D 抽象化モジュール (1 1 8) は、D 3 D アプリケーションから D 3 D コマンド (2 1 2) 及び D 3 D オブジェクト (2 1 4) を直接に受信するように構成され得る。

【 0 0 2 6 】

[0030] 一実装において、受信される D 3 D コマンド (2 1 2) は、高頻度のコマンドであって、クライアント (1 0 2) からの1つ以上のネットワークラウンドトリップを必要とする。そのようなネットワークラウンドトリップは、高頻度のコマンド実行時の遅れをもたらす。そのようなネットワークラウンドトリップを回避するために、D 3 D 抽象化モジュール (1 1 8) が、抽象化レイヤにおいて、D 3 D オブジェクトそれぞれにハンドル又は識別子を割り当てるように構成され得る。D 3 D 抽象化モジュール (1 1 8) がその後、クライアント端末上で D 3 D オブジェクトを生成するためのハンドルを、別のパラメータと一緒にデータストリーム (2 1 6) に記録する。ハンドルは、サーバー (1 0 6) のメモリー (1 1 4) 中の別のプログラムデータ (2 1 8) にストアされ得る。

【 0 0 2 7 】

[0031] D 3 D 抽象化モジュール (1 1 8) は、D 3 D コマンド (2 1 2) 及び D 3 D オブジェクト (2 1 4) をデータストリーム (2 1 6) の中に抽象化し、ネットワーク (1 0 4) を介しそれらを送信する。これに関連し、リモートアダプター (1 2 0) が、クライアント (1 0 2 - 1) のディスプレイアダプターの利用可能性及び機能に関する情報の問い合わせをする。クライアント (1 0 2 - 1) の D 3 D プラグイン (1 1 2) から利用可能性及び機能の確認を受信すると、リモートアダプター (1 2 0) は、D 3 D 抽象化モジュール (1 1 8) と D 3 D プラグイン (1 1 2) との間に通信チャネルを確立する。D 3 D 抽象化モジュール (1 1 8) は、通信チャネルを介し、抽象化した D 3 D コマンド (2 1 2) 及び D 3 D オブジェクト (2 1 4) をクライアント (1 0 2 - 1) の D 3 D プラグイン (1 1 2) に送信する。

【 0 0 2 8 】

[0032] 抽象化に加えて D 3 D 抽象化モジュール (1 1 8) は、リソースの集中型符号化を実行するように構成され得る。リソースの集中型符号化は、当技術分野で周知の D 3 D リソースを符号化するために利用可能な圧縮及びデシメーション方法を包含している。これは、複数のアプリケーション及び再接続に対して、サーバー (1 0 6) が効率的に帯域幅を使用できるようにする。例えば、D 3 D アプリケーションは、テクスチャマッピング、メッシュデータ、又は利用可能な帯域幅よりもサイズがより大きな通常のビットマップ画像などの D 3 D リソースを利用し得る。そのような場合、D 3 D 抽象化モジュール (1 1 8) は、リソースサイズを低減するための周知の圧縮及び/又はデシメーション方法を使用する。システムの広範囲に渡って持続するキャッシュが実行され、サーバー (1 0 6) の帯域幅を効率的に使用するためにリソース符号化プロセスが集中化され得る。

例示的クライアント

[0033] 図 3 は、3 D 画像のリモートレンダリングを実装するための例示的クライアント (1 0 2) を示す。例示的クライアント (1 0 2) は、任意のクライアント (1 0 2) 、例えば、クライアント (1 0 2 - 1) であり得、プロセッサ (3 0 2) 、ネットワークインターフェース (単数又は複数) (3 0 4) 、入力/出力インターフェース (単数又は複数) (3 0 6) 、メモリー (1 0 8) 、及び入力/出力装置 (単数又は複数) (3 1 0) を含む。ネットワークインターフェース (単数又は複数) (3 0 4) は、ネットワーク (1 0 4) を介しクライアント (1 0 2) 間の通信を容易にする。

【 0 0 2 9 】

[0034] 例示的実装において、ネットワークインターフェース (3 0 4) は、クライアン

10

20

30

40

50

ト(102-1)をサーバー(106)又はクライアント(102-1)以外のクライアント(102)に接続するための1つ以上のポートを含む。ネットワークインターフェース(単数又は複数)(304)は、有線ネットワーク(例えばLANネットワーク、ケーブルネットワークなど)並びに無線ネットワーク(例えばWLANネットワーク、セルラーネットワーク、衛星ネットワークなど)を含む様々なネットワークタイプ及びプロトコルタイプを用いた通信を容易にし得る。

【0030】

[0035]入力/出力インターフェース(単数又は複数)(306)は、例えば、スキャナーポート、マウスポート、キーボードポートなどを含んでいて、データをクライアント(102)から受信するか又はそれに送信し得る。一実装において、クライアント(102-1)の入力/出力インターフェース(単数又は複数)(306)は、例えば、ネットワーク(104)を介しD3Dコマンド(212)及びD3Dオブジェクト(214)などのデータを受信する。

10

【0031】

[0036]クライアント(102-1)の入力/出力インターフェース(単数又は複数)(306)は、同期化のためにレンダリングされるグラフィック画像を代表とするデータ又はコンテンツなどのデータをサーバー(106)に送信する。入力/出力インターフェース(単数又は複数)(306)は、入力/出力装置(310)によってレンダリングされたグラフィック画像などのデータを表示し得る。

【0032】

20

[0037]メモリー(108)は、ランダムアクセスメモリー(RAM)のような揮発性メモリー、及び/又は読み出し専用メモリー(ROM)、及び/又はフラッシュRAMのような不揮発性メモリー双方の形式の任意の計算機可読媒体であり得る。メモリー(108)は、プロセッサ(302)によってそれに即座にアクセス可能なデータ及び/又はその上で作動されるプログラムモジュールを含む。例示的実装において、メモリー(108)は、プログラムモジュール(110)及びプログラムデータ(308)を含む。プログラムモジュール(110)は、D3Dプラグイン(112)及びその他のモジュール(312)を含み得る。プログラムデータ(308)は、データストリーム(314)、グラフィック画像(316)、及びその他のプログラムデータ(318)を含むパラメータをストアし得る。

30

【0033】

[0038]D3Dプラグイン(112)は、D3Dコマンド(212)及びD3Dオブジェクト(214)を含むデータストリーム(314)を受信し、D3Dコマンド(212)を抽出する。更に、D3Dプラグイン(112)は、D3Dコマンド(212)を実行し、クライアント端末のGPUを使用することによって出力グラフィック画像をレンダリングする。一実装において、クライアント(102-1)のD3Dプラグイン(112)が、ネットワーク(104)を介しサーバーからデータストリーム(314)を受信するように構成され得る。D3Dプラグイン(112)は、当技術分野において周知の任意の解析方法を使用しデータストリーム(314)解析し、データストリーム(314)からD3Dコマンド(212)を抽出するように構成され得る。

40

【0034】

[0039]D3Dプラグイン(112)は、データストリーム(314)と一緒に受信したハンドルを使用しD3Dオブジェクト(214)を生成するように構成され得る。前述したように、ハンドルは、D3Dアプリケーションのようなグラフィックアプリケーションに関連付けられたリソースアドレスを含むエントリーを有していて、D3Dオブジェクト(214)などのオブジェクトが識別され、検査され、生成され、及び/又は変更され得る。従ってD3Dオブジェクトは、ハンドルを割り当てられていて、D3Dプラグイン(112)によってクライアント端末において実際のD3Dオブジェクト(214)を生成し得る。D3Dプラグイン(112)が、その後、D3Dコマンド(212)を実行し、クライアントのGPUを使用しグラフィック画像(316)をレンダリングし得る。グラ

50

フィック画像(316)のレンダリングは、当技術分野で周知のレンダリング方法によって実行され得る。

【0035】

[0040]レンダリングに加えてD3Dプラグイン(112)は、サーバー(106)のデスクトップ上に表示されるウィンドウ形状における変化をトラッキングし、サーバーのデスクトップ表示に従って、クライアント(102-1)において変化するように構成される。ウィンドウ形状の変化は、ウィンドウのリサイズ、ウィンドウの移動若しくは位置、及び/又はサーバー(106)のデスクトップの可視領域の変化を含み得る。

【0036】

[0041]ウィンドウ形状の変化を支援するためのプロキシウィンドウが、クライアント(102-1)において使用され得、サーバー(106)のウィンドウにおいて発生する幾何学的変化を再生成する。プロキシウィンドウは、サーバー(106)におけるウィンドウ及びクライアント(102)におけるウィンドウのような、同一セッションにおいて起動された任意のウィンドウとの相互運用性を保証する。プロキシウィンドウは、例えばデスクトップ構成機能が利用できないとき、クライアント(102)において確立され得る。デスクトップ構成は、クライアントのディスプレイ画面上にグラフィック画像の画素が表示される方法を変更する機能であって、サーバー(106)のデスクトップ・ウィンドウ・マネージャーによって制御され得る。

【0037】

[0042]D3Dプラグイン(112)は、クライアントサイド(102)、例えばクライアント(102-1)上にDirect3Dグラフィックをフルスクリーン占有モードで提示するようにも構成される。フルスクリーン表示占有モードは、ユーザーから介入をせずに、画面解像度及びピクセルフォーマットを自動的にキャプチャし、設定/リストアする。クライアントウィンドウは、画像を表示するために使用される簡素なウィンドウであり得る。しかしながら、Direct3Dアプリケーションは、簡素なウィンドウフォーマットによって支援され得ないフルスクリーン表示占有モード下で実行する。D3Dプラグイン(112)は、クライアントのウィンドウにフルスクリーン表示を用いたフルスクリーン占有モードのグラフィックを提示し、ユーザー体験を改善する。

【0038】

[0043]D3Dプラグイン(112)は、サーバーが同期化するための情報を必要とする場合、レンダリングされたグラフィック画像(316)に関連する情報をサーバー(106)に送信するように構成される。一実装において、D3Dプラグイン(112)は、クライアントサイド専用のレンダリングプロトコルに基づいてクライアント(102-1)のウィンドウに出力グラフィック画像をレンダリングする。そのような場合、レンダリングされたコンテンツが、デフォルトによってクライアント(102)にストアされていて、また、サーバーがコンテンツを必要とする場合、サーバー(106)に送信され得る。従って、D3Dプラグイン(112)は更に、認識済のサーバー(106)の不可視のデスクトップと、サーバー(106)のデスクトップに関するクライアントサイド(102)の可視表示との間のトラッキング及び同期化が可能ないように構成され得る。

【0039】

[0044]更に、同期化及びコンテンツの一貫性を達成するためにD3Dプラグイン(112)は、サーバー(106)から発行される様々なDirect3Dコマンド(212)と、クライアント(102)のデスクトップに実行される読み出しアクセス/書き込みアクセスと、をトラッキングするように構成され得る。D3Dプラグイン(112)は更に、サーバー(106)及びクライアント(102-1)などのクライアント(102)双方によって送信/受信される画素それぞれに関する更新情報を維持し得る。例えば、デスクトップ構成が利用できないとき同期化するために、画素に関する情報が更新され得る。

例示的アーキテクチャ

[0045]図4は、Direct3D(D3D)遠隔化アーキテクチャ(400)の例示的実装を示している。アーキテクチャ(400)は、D3D遠隔化アーキテクチャの論理的

10

20

30

40

50

ビューであって、リモートセッション内でグラフィック画像をレンダリングするために使用される関連コンポーネントを表示する。アーキテクチャ(400)は、サーバーサイド(402)(例えばサーバー(106))において作動するコンポーネントグループを含み得る。サーバーサイド(402)のコンポーネントは、D3DアプリケーションからD3Dコマンドを受信し、ネットワーク上のクライアントサイド(404)(例えばクライアント(102))に送信されるストリームデータとしてD3Dコマンド及びD3Dオブジェクトを抽象化する。クライアントサイド(404)は、クライアント(102-1)などのクライアント(102)と関係するコンポーネントグループを含む。クライアントサイド(404)のコンポーネントが、データストリームを受信し、D3Dコマンドを抽出し、実際のD3Dオブジェクトを生成し、D3Dコマンドを実行し、クライアント端末において、出力グラフィック画像をレンダリングする。本アーキテクチャ(400)は更に、サーバー(106)又はクライアント(102)によってセッション空間(406)として参照される同一セッション内においてアクセスされるコンポーネントと、サーバー(106)又はクライアント(102)によってグローバル空間(408)として参照される別のセッション内においてアクセスされるコンポーネントと、を含む。

10

【0040】

[0046]一実装において、サーバーサイド(402)のD3Dアプリケーション(410)が、ユーザーからの入力として1つ以上のD3Dコマンド(212)を受信する。D3Dアプリケーション(410)は、ゲームアプリケーションのような複雑な計算を含む高性能グラフィックアプリケーションであり得る。そのようなアプリケーションに対しD3Dコマンド(212)及びD3Dオブジェクト(214)は、Terminal Server(登録商標)TS/D3D抽象化レイヤ(412)においてデータストリーム(216)として抽象化される。一実装において、D3D抽象化モジュール(118)が、Terminal Server(登録商標)TS/D3D抽象化レイヤ(412)上でオブジェクトを抽象化するためのコマンドを実行する。更に、D3D抽象化モジュール(118)が1つ以上のD3Dオブジェクト(214)にハンドルを割り当て、実際のD3Dオブジェクト(214)がクライアント端末において、ハンドルを使用し生成され得る。

20

【0041】

[0047]D3Dコマンド(212)及びD3Dオブジェクトの抽象化に関し、D3D抽象化モジュール(118)は、D3DAPI(414)と通信するが、場合によってはGPUが利用可能である場合、アプリケーションは、サービスサイドにGPU(すなわちローカルGPUドライバー(416))を実装し得る。この場合、(又はOSポリシーごとのデフォルトで)GPUドライバー(416)が、サーバー(106)にローカルに存在し得、Win32kプロセス(418)と通信し得る。言い換えると、クライアント(例えばクライアント(102-1)などのクライアント(102))においてリモートGPUを提示しているリモートアダプター(120)は、存在しているローカルGPUドライバー(416)と協力し実行され得るが、ローカルGPUドライバー(416)は、この遠隔化アーキテクチャが機能するように存在し得ない。ローカルGPUドライバー(416)は、遠隔化アーキテクチャが既存のローカルグラフィックアーキテクチャとどのように統合されるか示す。

30

40

【0042】

[0048]Win32kプロセス(418)は、メモリー、プロセッサ(単数又は複数)、及び入力/出力装置のような計算リソースに対し最小レベルの抽象化を提供するWindows(登録商標)オペレーティングシステムのカーネルプロセスである。Win32kプロセス(418)は、グラフィック関連タスクを実行するグラフィック・デバイス・インターフェース(GDI)のGDIアプリケーション(420)と通信する。GDIアプリケーション(420)は、グラフィックオブジェクトを表示し、モニター及びプリンターなどの出力装置にグラフィックオブジェクトを送信するために使用されるMicrosoft Windows(登録商標)オペレーティングシステムのインターフェースであり得る。GDIアプリケーション(420)は、直線、曲線、及びレンダリ

50

ングフォントなどを描画するようなグラフィック関連タスクを実行し得る。

【0043】

[0049]一実装において、そのようなグラフィック関連タスクを実行するためにGDIアプリケーション(420)は、GDI32プロセス(422)を使用する。GDI32プロセス(422)は、サーバーのWindows(登録商標)オペレーティングシステムのGDIに関する機能を含んでいて、クライアント(102)のディスプレイウィンドウ上に画像を表示するためのオブジェクト及びプロセス情報を生成する。GDI32プロセス(422)は、D3Dアプリケーション(410)に対しD3Dオブジェクト(214)の生成のようなグラフィック関連機能を実行する。

【0044】

[0050]D3Dオブジェクト(214)の生成後、D3Dコマンド(212)及びD3Dオブジェクト(214)は、D3Dアプリケーション(410)及びD3DAPI(414)に連絡される。連絡の間、D3D抽象化レイヤ(412)において、D3D抽象化モジュール(118)が、D3Dコマンド(212)及びD3Dオブジェクト(214)をデータストリーム(216)の中に抽象化する。D3D抽象化モジュール(118)は、D3Dオブジェクト(214)のハンドルもデータストリーム(216)の中に抽象化する。

【0045】

[0051]D3D抽象化モジュール(118)は、ネットワーク(104)を介し、クライアント(102)にデータストリーム(216)を送信する。これに関連し、リモートアダプター(120)は、クライアント(102-1)などのクライアント(102)のディスプレイアダプターの利用可能性及び機能に関する問い合わせをする。クライアント(102-1)のD3Dプラグイン(112)から確認を受信すると、リモートアダプター(120)は、ネットワーク(104)を介しデータストリーム(216)を送信するための通信チャネルを確立する。

【0046】

[0052]クライアントサイド(404)において、D3Dプラグイン(112)が、リモートアダプター(120)によって確立される通信チャネルを介しデータストリーム(314)を受信する。D3Dプラグイン(112)によって受信されるデータストリーム(314)は、D3Dコマンド(212)、D3Dオブジェクト(214)、及び実際のD3Dオブジェクト(214)を再生成するためのハンドルを含む。D3Dプラグイン(112)は、当技術分野で周知の任意の解析方法を使用し、受信したデータストリーム(314)を解析し、D3Dコマンド(212)及びD3Dオブジェクト(214)を抽出する。

【0047】

[0053]D3Dプラグイン(112)は、その後、D3Dコマンド(212)を実行し、D3Dアプリケーション(410)に関連付けられたグラフィックを表示する。これに関連し、D3Dプラグイン(112)は、クライアント(102-1)などのクライアント(102)のD3DAPI(424)と通信する。D3DAPI(424)は、D3Dプラグイン(112)とグラフィックハードウェアとの間においてD3Dコマンド(212)及びD3Dオブジェクト(214)を通信し、グラフィックアプリケーションを実行する。D3DAPI(424)は、グラフィックを表示するために使用され得るグラフィック画像をレンダリングするコマンドも含む。

【0048】

[0054]D3DAPI(424)は更に、ハンドルを使用した実際のD3Dオブジェクト(214)生成のようなグラフィック関連タスクを実行するために、Win32kプロセス(426)とインタラクションする。Win32kプロセス(426)(Windows(登録商標)オペレーティングシステムのカーネルプロセス)は、サーバー(106)のWin32kプロセス(418)と同様であり得る。実際のD3Dオブジェクト(214)の生成に関しWin32kプロセス(426)は、実際のD3Dオブジェクト(21

10

20

30

40

50

4) を生成するグラフィック・デバイス・インターフェースプロセスのGDI32プロセス(428)とインタラクションする。

【0049】

[0055]実際のD3Dオブジェクト(214)の生成後、クライアントサイド(404)のGDI32プロセス(428)は、D3DAPI(424)を介しD3Dプラグイン(112)にD3Dオブジェクト(214)を連絡する。更に、D3Dプラグイン(112)は、D3Dオブジェクト(214)を使用し、D3Dコマンド(212)を実行し、グラフィック画像(316)をレンダリングする。D3Dプラグイン(112)は、GPUドライバと同様にグラフィック画像(316)をレンダリングする。レンダリングされたグラフィック画像(316)は、その後、クライアント(102-1)の入力/出力装置(310)を介し表示される。実装において、クライアント(102)それぞれは、リモートターミナルサーバクライアント(430)であり得る。クライアント(102)のD3Dプラグインは、クライアント(102)が切断され、後で再接続しようとするときも、レンダリングプロセス中のセッションにおいてD3Dオブジェクト(214)を持続し得る。再接続の成功後、サーバ(106)及びクライアント(102)は、D3Dアプリケーション(410)の実行を再開する。

10

【0050】

[0056]実装において、D3Dアプリケーション(410)のグラフィック表示は、1つ以上のクライアント(102)によって共有され得るテクスチャマッピング、ノーマルマッピングなどのような1つ以上のD3Dリソースを使用することを含む。リソースを共有するためにD3Dプラグイン(112)は、D3Dリソースそれぞれにハンドル又は識別子を割り当てるように構成され得るが、ハンドルの使用は、1つ以上のクライアント(102)からのネットワークラウンドトリップを必要とする。絶え間のないネットワークラウンドトリップを回避するために、集中型サーバコンポーネントは、クライアント(102)に割り当てられたプロキシとして使用されるハンドルを提供する。例えば、RDPDD(Remote Desktop Protocol Device Driver)(432)又はRDPWD(Remote Desktop Protocol Win32k Driver)(434)がそれぞれ、クライアント(102)に対するプロキシとしてセッション空間(406)又はグローバル空間(408)において使用され得る。

20

30

【0051】

[0057]更に、累進的なテクスチャレンダリング方法、フレームレート制御方法及び自動再接続方法のような1つ以上の方法が、D3D遠隔化アーキテクチャにおいて実装され得る。一実装において、D3Dアプリケーション(410)は、より良いエンドユーザー体験を達成するために、より多くのネットワーク回線容量を消費するメッシュ及びテクスチャなどのD3Dリソースも処理する。そのような場合、累進的なテクスチャレンダリング方法が効率的に帯域幅を割り当て、より良いエンドユーザー体験を達成するように構成され得る。

【0052】

[0058]更に、D3D抽象化モジュール(118)が、D3Dアプリケーション(410)のタイプに基づいてフレーム転送レートを制限することによって、サーバ(106)が帯域幅消費を規制可能なフレームレート制御方法を使用し得る。そのようなフレームレート制御方法は、複数の接続並びにアプリケーション間の拡張性及び一貫性を改善し得る。

40

【0053】

[0059]加えてD3Dアプリケーション(410)からシームレスなリモート自動再接続体験を提供する自動再接続方法が実装され得る。例えば、クライアント(102)がD3Dアプリケーション(410)の実行中に切断されたとき、実行中のD3Dアプリケーション(410)を支援可能な新しいクライアントが接続され得る。一実装において、新しいクライアントの再接続中、D3Dアプリケーション(410)は、DEVICE_LO

50

ST又はDEVICE_REMOVEDなどのメッセージを受信する。D3D抽象化モジュール(118)は、その後、D3Dアプリケーション(410)を切断前のステータスにリストアし、D3Dアプリケーション(410)実行をレジュームする。別の実装において、新しいクライアントに再接続中、D3D抽象化モジュール(118)は、サーバーサイド(402)において、オブジェクトコンテンツを再生成するために必要とされるD3Dコマンド(212)及びD3Dオブジェクトの完全な抽象化を実行する。

【0054】

[0060]サーバー(106)は、切断セッション中、無限数のD3Dコマンド(212)を蓄積しないことを保証するための更なる方法が使用され得る。本実装において、サーバー(106)において、無限数のD3Dコマンド(212)を回避するために、D3Dアプリケーション(410)を呼び出す1つ以上のカーネルのスレッドが遮断され得る。

10

例示的プロセス

[0061]Direct3D遠隔化アーキテクチャを実装する例示的方法を以下に説明する。これらの方法のいくつか又はすべては必要でないが、図1~4に示されるようなアーキテクチャの少なくとも一部は実装され得る。また、本方法のいくつかの動作は、説明した順番で実行される必要性はなく、状況によって変更され、及び/又は完全に省略されることを理解されたい。

【0055】

[0062]図5は、サーバー端末におけるD3Dグラフィックアプリケーション中の抽象化コマンドに関する例示的プロセス(500)を示している。例示的プロセス(500)が説明される順番は、制限として理解されるように意図されておらず、説明したどのような方法のブロックも本方法又は代替方法を実装するために順不同に接続され得る。更に、個々のブロックは、本明細書に記載した対象項目の趣旨及び範囲から逸脱せずに本方法から削除され得る。更に、本方法は、適切な任意のハードウェア、ソフトウェア、ファームウェア、又はその組み合わせで実装され得る。

20

【0056】

[0063]ブロック(502)において、1つ以上のグラフィックアプリケーションコマンドが識別され、グラフィックアプリケーションから受信され得る。D3D抽象化モジュール(118)は、D3D抽象化レイヤ(412)においてグラフィックアプリケーションコマンドを監視していて、D3Dアプリケーション(410)から受信し得る。

30

【0057】

[0064]ブロック(504)において、アプリケーションコマンドが、APIを介しグラフィックハードウェアドライバに連絡される。D3D抽象化モジュール(118)は、D3D API(414)を介し、受信したコマンド(212)をグラフィックハードウェアドライバに連絡し得る。

【0058】

[0065]ブロック(506)において、グラフィックハードウェアドライバによって生成されたアプリケーションコマンド及びオブジェクトが、データストリームの中に抽象化される。D3D抽象化モジュール(118)が、生成されたD3Dコマンド(212)及びD3Dオブジェクト(214)をデータストリーム(216)の中に隠されたデータとして抽象化し得、それらがネットワーク(104)上に送信される。更に、D3D抽象化モジュール(118)が、1つ以上のD3Dオブジェクトにハンドル又は識別子を割り当てる。D3D抽象化モジュール(118)は、クライアント(102-1)においてD3Dオブジェクト(214)を生成するために使用され得るハンドルを別のパラメータと一緒にデータストリーム(216)の中に記録する。

40

【0059】

[0066]ブロック(508)において、データストリームが、クライアントにネットワークを介し送信される。D3D抽象化モジュール(118)は、1つ以上のクライアント(102)、例えばクライアント(102-1)にデータストリーム(216)を送信し得る。データストリーム(216)がその後、クライアント(102-1)のD3Dプラグ

50

イン(112)によって、データストリーム(314)として受信される。

【0060】

[0067]一実装において、データストリーム(216)中のD3Dコマンド(212)及びD3Dオブジェクト(214)は、当技術分野で周知の任意の圧縮及びデシメーション方法を使用し圧縮され得る。例えば、D3Dリソースが、テクスチャ、メッシュデータなどのようにサイズが大きく、ネットワーク(104)の帯域幅に適合することが不可能な場合、そのような技法が使用され得る。圧縮されたD3Dリソースは、その後、リモートアダプター(120)を介しネットワーク(104)上に送信され得る。

【0061】

[0068]リモートアダプター(120)は、D3D抽象化モジュール(118)をクライアント(102-1)のD3Dプラグイン(112)と通信可能にする。これに関連し、リモートアダプター(120)は、クライアント(102-1)のディスプレイアダプターの利用可能性及び機能ステータスをチェックする。確認がD3Dプラグイン(112)から一旦、受信されると、リモートアダプター(120)は、D3D抽象化レイヤ(412)とD3Dプラグイン(112)との間に通信チャネルを確立する。D3D抽象化モジュール(118)は、確立した通信チャネルを介し、ネットワーク(108)上のクライアント(102)にデータストリーム(216)を送信する。

【0062】

[0069]図6は、3D画像をレンダリングするための例示的プロセスを示す。例示的プロセス(600)が説明されている順番は、制限として理解されるように意図されておらず、本方法又は代替方法を実装するために説明した方法のブロックがどれでも順不同に接続され得る。更に、個々のブロックは、本明細書に記載した対象項目の趣旨及び範囲から逸脱せず本方法から削除され得る。更に、本方法は、適切な任意のハードウェア、ソフトウェア、ファームウェア、又はその組み合わせにて実装され得る。

【0063】

[0070]ブロック(602)において、データストリームがサーバーから受信される。クライアント(102-1)のD3Dプラグイン(112)が、サーバー(106)のD3D抽象化モジュール(118)からデータストリーム(314)を受信する。

【0064】

[0071]ブロック(604)において、D3Dコマンド及びD3Dオブジェクトが、データストリームから抽出される。D3Dプラグイン(112)が、データストリーム(314)からD3Dコマンド(212)及びD3Dオブジェクト(214)を抽出し得る。更に、D3Dプラグイン(112)は、クライアント(102-1)のGPUの支援を用いてD3Dコマンド(212)を実行し得る。

【0065】

[0072]ブロック(606)において、グラフィック画像形式の出力は、クライアントにおける入力/出力装置を介しレンダリングされる。3Dグラフィック画像(316)のようなD3Dアプリケーション(410)の出力は、当技術分野で周知の任意のレンダリングアルゴリズムを使用し、クライアント(102-1)の入力/出力装置(310)を介し、レンダリングされ得る。

【0066】

[0073]前述した任意の方法に関する動作のいくつかはプロセッサ又はその他の計算装置によって、1つ以上の計算機可読媒体上にストアされている命令に基づいて実装される。計算機可読媒体は、リソースモデリングアプリケーションによって利用可能なローカル又はリモートにアクセスされ得る任意の媒体であり得る。非限定の例として計算機可読媒体は、計算機記憶媒体及び通信媒体を含む。計算機記憶媒体は、計算機読込可能命令、データ構造、プログラムモジュール、又はその他のデータなどの情報記憶に関する任意の方法若しくは技術で実装された揮発性及び不揮発性並びに取り外し可能及び取り外し不可能媒体を含む。計算機記憶媒体は、RAM、ROM、EEPROM、フラッシュメモリー、若しくはその他のメモリー技術、CD-ROM、デジタル多用途ディスク(DVD)、若

10

20

30

40

50

しくはその他の光記憶装置、磁気カセット、磁気テープ、磁気ディスク記憶装置、若しくはその他の磁気記憶装置、又は所望した情報をストアするために使用され、アプリケーションをモデル化するリソースによってアクセスされ得る別の任意の媒体、を含むがこれらに限定しない。通信媒体は、通常、任意の情報伝達媒体に対する計算機可読命令、データ構造、プログラムモジュール、又はその他のデータを具体化する。いくつかの上記の組み合わせも計算機可読媒体の範囲内に含まなくてはならない。

【 0 0 6 7 】

結論

[0074]本発明は、自己記述型データレンダリングを実装するための構造的特徴及び/又は方法論の動作に対し特定の言語で説明しているが、本発明が必ずしも説明した特定の特徴又は動作に限定されるわけではないことが理解されよう。むしろ、特定の特徴及び動作は、本発明を実装する例示的形式として開示されている。

10

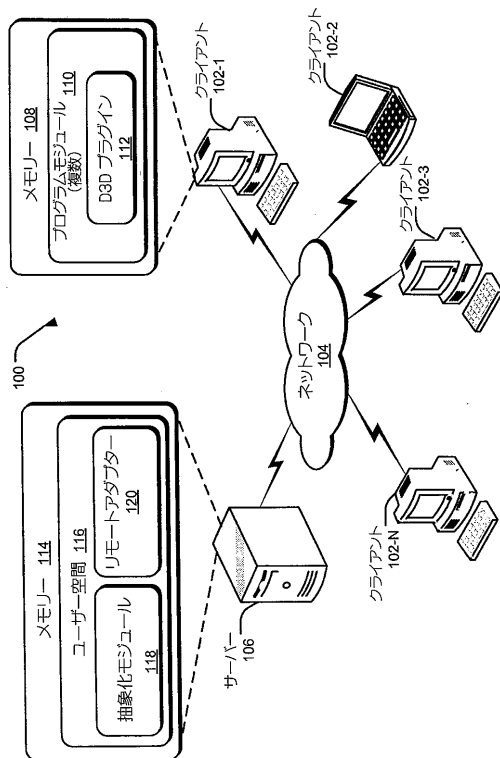
【符号の説明】

【 0 0 6 8 】

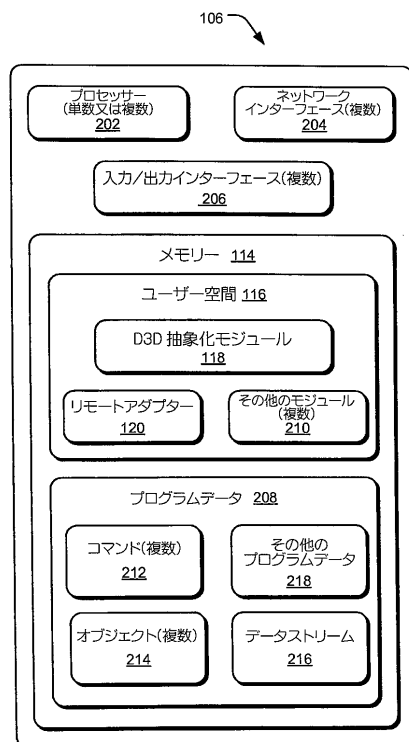
1 0 0	例示的ネットワーク計算環境	
1 0 2	クライアント	
1 0 2 - 1	クライアント	
1 0 2 - 2	クライアント	
1 0 2 - 3	クライアント	
1 0 2 - N	クライアント	20
1 0 4	ネットワーク	
1 0 6	サーバー	
1 0 8	メモリー	
1 1 0	プログラムモジュール	
1 1 2	D 3 D プラグイン	
1 1 4	メモリー	
1 1 6	ユーザー空間	
1 1 8	D 3 D 抽象化モジュール	
1 2 0	リモートアダプター	
2 0 2	プロセッサ	30
2 0 4	ネットワークインターフェース	
2 0 6	入力/出力インターフェース	
2 0 8	プログラムデータ	
2 1 0	その他のモジュール	
2 1 2	D 3 D コマンド	
2 1 4	D 3 D オブジェクト	
2 1 6	データストリーム	
2 1 8	その他のプログラムデータ	
3 0 2	プロセッサ	
3 0 4	ネットワークインターフェース	40
3 0 6	入力/出力インターフェース	
3 0 8	プログラムデータ	
3 1 0	入力/出力装置	
3 1 2	その他のモジュール	
3 1 4	データストリーム	
3 1 6	グラフィック画像	
3 1 8	その他のプログラムデータ	
4 0 0	遠隔化アーキテクチャ	
4 0 2	サーバーサイド	
4 0 4	クライアントサイド	50

- 4 0 6 セッション空間
- 4 0 8 グローバル空間
- 4 1 0 D 3 Dアプリケーション
- 4 1 2 D 3 D抽象化レイヤ
- 4 1 4 D 3 Dアプリケーションプログラミングインターフェース
- 4 1 6 ローカルGPUドライバー
- 4 1 8 Win32kプロセス
- 4 2 0 GDIアプリケーション
- 4 2 2 GDI32プロセス
- 4 2 4 D 3 Dアプリケーションプログラミングインターフェース
- 4 2 6 Win32kプロセス
- 4 2 8 GDI32プロセス
- 4 3 0 リモートターミナルサーバクライアント
- 4 3 2 RDPDD (リモート・デスクトップ・プロトコル・デバイス・ドライバ
- ー)
- 4 3 4 RDPWD (リモート・デスクトップ・プロトコル・Win32k・ドラ
- イバー)

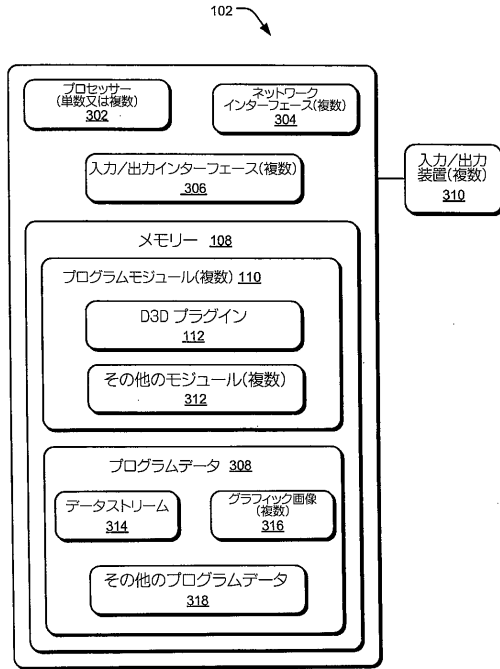
【図1】



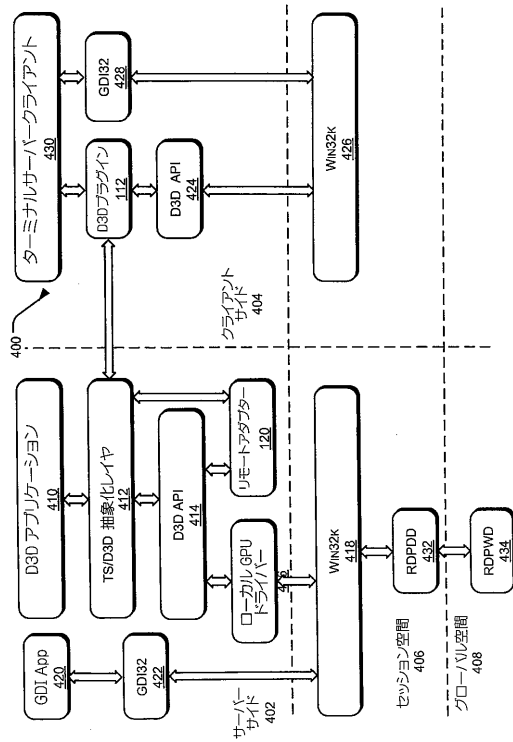
【図2】



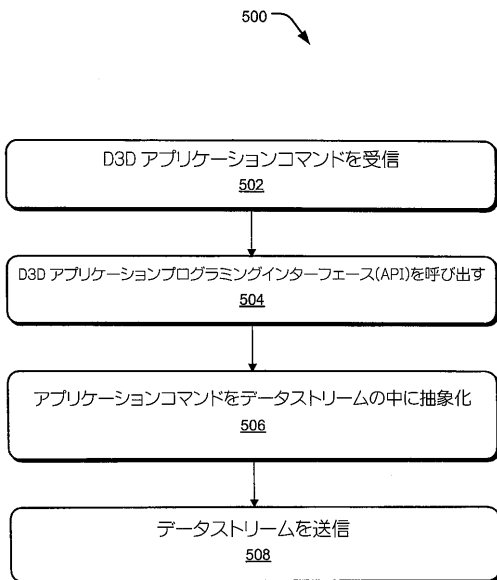
【図3】



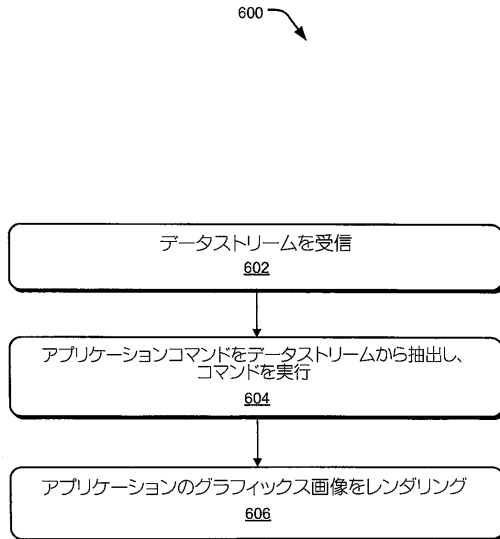
【図4】



【図5】



【図6】



フロントページの続き

(74)代理人 100153028

弁理士 上田 忠

(72)発明者 チウ, カン

アメリカ合衆国ワシントン州 9 8 0 5 2 , 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, インターナショナル・パテンツ

(72)発明者 アブド, ナディム・ワイ

アメリカ合衆国ワシントン州 9 8 0 5 2 , 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, インターナショナル・パテンツ

審査官 千葉 久博

- (56)参考文献 特開 2 0 0 7 - 0 0 4 2 2 8 (J P , A)
特開 2 0 0 6 - 2 3 6 3 3 4 (J P , A)
特開 2 0 0 5 - 2 8 4 6 9 4 (J P , A)
特開 2 0 0 5 - 0 1 1 3 5 4 (J P , A)
特開 2 0 0 2 - 0 9 2 6 2 4 (J P , A)
特開 2 0 0 2 - 0 6 3 6 0 1 (J P , A)
特開平 1 1 - 0 6 6 1 0 6 (J P , A)
特表 2 0 0 7 - 5 1 0 2 0 2 (J P , A)
特表 2 0 0 2 - 5 0 5 7 7 6 (J P , A)
米国特許出願公開第 2 0 0 7 / 0 0 6 1 8 7 8 (U S , A 1)
米国特許第 0 5 4 8 7 1 7 2 (U S , A)
米国特許第 0 7 1 6 2 5 2 8 (U S , B 1)
米国特許第 0 6 7 1 4 2 0 0 (U S , B 1)
韓国公開特許第 1 0 - 2 0 0 6 - 0 0 6 7 2 4 8 (K R , A)

(58)調査した分野(Int.Cl., DB名)

G 0 6 T 1 1 / 0 0 - 1 1 / 4 0 , 1 5 / 0 0 - 1 5 / 8 7

G 0 6 F 1 3 / 0 0

G 0 6 T 1 / 0 0 , 1 3 / 0 0 - 1 3 / 8 0 , 1 9 / 0 0 , 1 9 / 2 0