

(19) 日本国特許庁(JP)

(12) 公表特許公報(A)

(11) 特許出願公表番号

特表2008-538056

(P2008-538056A)

(43) 公表日 平成20年10月2日(2008.10.2)

(51) Int.Cl.
H04L 12/26 (2006.01)F I
H04L 12/26テーマコード (参考)
5K030

審査請求 未請求 予備審査請求 未請求 (全 55 頁)

(21) 出願番号 特願2007-551333 (P2007-551333)
 (86) (22) 出願日 平成18年1月11日 (2006.1.11)
 (85) 翻訳文提出日 平成19年9月12日 (2007.9.12)
 (86) 国際出願番号 PCT/US2006/000887
 (87) 国際公開番号 W02006/076388
 (87) 国際公開日 平成18年7月20日 (2006.7.20)
 (31) 優先権主張番号 11/035,511
 (32) 優先日 平成17年1月14日 (2005.1.14)
 (33) 優先権主張国 米国 (US)
 (31) 優先権主張番号 11/036,489
 (32) 優先日 平成17年1月14日 (2005.1.14)
 (33) 優先権主張国 米国 (US)
 (31) 優先権主張番号 11/036,840
 (32) 優先日 平成17年1月14日 (2005.1.14)
 (33) 優先権主張国 米国 (US)

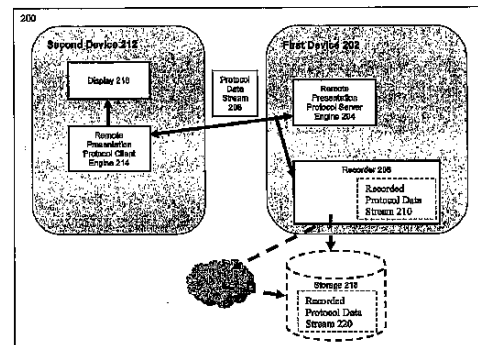
(71) 出願人 502239313
 サイトリックス システムズ, インコー
 ポレイテッド
 アメリカ合衆国 フロリダ 33309,
 フォート ローダーデール, ダブリュー
 ー, サイプレス クリーク ロード 8
 51
 (74) 代理人 100078282
 弁理士 山本 秀策
 (74) 代理人 100062409
 弁理士 安村 高明
 (74) 代理人 100113413
 弁理士 森下 夏樹

最終頁に続く

(54) 【発明の名称】 記録されたコンピュータセッションに対するプレイバック命令を生成する方法およびシステム

(57) 【要約】

第1のデバイス上で実行するプロトコルエンジンは、第2のデバイス上で実行されるアプリケーションプログラムによって生成されたディスプレイデータを表す、複数のパケットを含む記録されたストリームを受信する。プロトコルエンジンは、記録されたストリームにおけるパケットに対して、人間が感知することが可能な態様でパケットのコンテンツをレンダリングすることを決定する。プロトコルエンジンは、プレイバックデータ構造において決定を格納する。パケットは、パケットによって格納された入力のタイプに応答してマーク付けされる。パケットのコンテンツをレンダリングする行先は、プレイバックデータ構造において格納され、マーク付けに応答する。



【特許請求の範囲】**【請求項 1】**

記録されたコンピュータセッションのプレイバックのためのプレイバック命令を生成する方法であって、該方法は、

(a) 第 1 のデバイス上で実行するプロトコルエンジンによって、記録されたストリームを受信するステップであって、該記録されたストリームは、第 2 のデバイス上で実行されるアプリケーションプログラムによって生成されるディスプレイデータを表す複数のパケットを含むステップと、

(b) 該プロトコルエンジンによって、該記録されたストリームにおけるパケットに対して、人間が感知することが可能な態様で該パケットのコンテンツをレンダリングすることを決定するステップと、

(c) 該プロトコルエンジンによって、プレイバックデータ構造における該決定を格納するステップと

を包含する、方法。

【請求項 2】

前記ステップ (b) は、入力フォーカスを有するアプリケーションプログラムの指示に応答して、少なくとも一つのパケットをレンダリングすることを決定することをさらに包含する、請求項 1 に記載の方法。

【請求項 3】

前記ステップ (b) は、前記少なくとも一つのパケットに格納されたユーザ入力のタイプの評価に応答して、該少なくとも一つのパケットをレンダリングすることを決定することをさらに包含する、請求項 1 に記載の方法。

【請求項 4】

前記ステップ (b) は、前記少なくとも一つのパケットによって格納されたグラフィックの更新のタイプの評価に応答して、該少なくとも一つのパケットをレンダリングすることを決定することをさらに包含する、請求項 1 に記載の方法。

【請求項 5】

前記ステップ (b) は、前記少なくとも一つのパケットによって格納された相互作用シーケンスのタイプの評価に応答して、該少なくとも一つのパケットをレンダリングすることを決定することをさらに包含する、請求項 1 に記載の方法。

【請求項 6】

記録されたコンピュータセッションのプレイバックの方法であって、該方法は、

(a) 第 1 のデバイス上で実行するバックグラウンドプロトコルエンジンによって、記録されたストリームを受信するステップであって、該記録されたストリームは、第 2 のデバイス上で実行するアプリケーションプログラムによって生成されたディスプレイデータを表す複数のパケットを含むステップと、

(b) 該バックグラウンドプロトコルエンジンによって、該記録されたストリームにおける少なくとも一つのパケットに対して、人間が感知することが可能な態様で該パケットをレンダリングすることを決定するステップと、

(c) 該バックグラウンドプロトコルエンジンによって、プレイバックデータ構造における決定を格納するステップと、

(d) フォアグラウンドプロトコルエンジンによって、該記録されたストリームから少なくとも一つのパケットを検索するステップと、

(e) 該フォアグラウンドプロトコルエンジンによって、該プレイバックデータ構造にアクセスするステップと、

(f) 該フォアグラウンドプロトコルエンジンによって、該プレイバックデータ構造に応答する、該少なくとも一つのパケットをレンダリングするステップと

を包含する、方法。

【請求項 7】

前記ステップ (b) は、入力フォーカスを有するアプリケーションプログラムの指示に

10

20

30

40

50

応答する、前記少なくとも一つの packets をレンダリングすることを決定することをさらに包含する、請求項 6 に記載の方法。

【請求項 8】

前記ステップ (b) は、前記少なくとも一つの packets に格納されたユーザ入力のタイプの評価に応答する、該少なくとも一つの packets をレンダリングすることを決定することをさらに包含する、請求項 6 に記載の方法。

【請求項 9】

前記ステップ (b) は、前記少なくとも一つの packets によって格納されたグラフィックの更新のタイプの評価に応答する、該少なくとも一つの packets をレンダリングすることを決定することをさらに包含する、請求項 6 に記載の方法。

10

【請求項 10】

前記ステップ (b) は、前記少なくとも一つの packets によって格納された相互作用シーケンスのタイプの評価に応答する、該少なくとも一つの packets をレンダリングすることを決定することをさらに包含する、請求項 6 に記載の方法。

【請求項 11】

前記ステップ (c) は、前記バックグラウンドプロトコルエンジンによって、前記プレイバックデータ構造において、バッファに前記 packets をレンダリングするための命令を格納することをさらに包含する、請求項 6 に記載の方法。

【請求項 12】

前記ステップ (f) は、前記フォアグラウンドプロトコルエンジンによって、バッファに前記少なくとも一つの packets のコンテンツをレンダリングすることをさらに包含する、請求項 6 に記載の方法。

20

【請求項 13】

記録されたセッションをレンダリングするシステムであって、該システムは、

第 1 のデバイス上で実行し、記録されたストリームを受信することに応答してプレイバックデータ構造を生成するバックグラウンドプロトコルエンジンであって、該記録されたストリームは、第 2 のデバイス上で実行されたアプリケーションプログラムによって生成されたディスプレイデータを表す、バックグラウンドプロトコルエンジンと、

該記録されたストリームを受信し、該バックグラウンドプロトコルエンジンによって生成された該プレイバックデータ構造に応答する該記録されたストリームをレンダリングする、フォアグラウンドプロトコルエンジンと

30

を備えた、システム。

【請求項 14】

前記バックグラウンドプロトコルエンジンは、前記記録されたストリームをレンダリングする前記フォアグラウンドプロトコルエンジンと実質的に同時に前記プレイバックデータ構造を生成する、請求項 13 に記載のシステム。

【請求項 15】

前記フォアグラウンドプロトコルエンジンは、前記第 1 のデバイスに存在する、請求項 13 に記載のシステム。

【請求項 16】

40

前記バックグラウンドプロトコルエンジンは、前記記録されたストリームにおける少なくとも一つの packets をレンダリングするための少なくとも一つの命令を前記プレイバックデータ構造に格納することをさらに包含する、請求項 13 に記載のシステム。

【請求項 17】

前記バックグラウンドプロトコルエンジンは、前記プレイバックデータ構造にメタデータを格納することをさらに包含する、請求項 13 に記載のシステム。

【請求項 18】

前記バックグラウンドプロトコルエンジンは、前記プレイバックデータ構造に、前記記録されたセッションにおける少なくとも一つの packets をレンダリングするための時間を示す記録を格納することをさらに包含する、請求項 13 に記載のシステム。

50

【請求項 19】

前記フォアグラウンドプロトコルエンジンは、前記記録されたセッションにおける少なくとも一つの packets を、人間が感知することが可能な態様で、バッファにレンダリングする、請求項 13 に記載のシステム。

【請求項 20】

前記フォアグラウンドプロトコルエンジンは、前記記録されたセッションにおける少なくとも一つの packets を、バッファにレンダリングする、請求項 13 に記載のシステム。

【請求項 21】

記録されたセッションをレンダリングするプレイバック命令を生成する方法であって、該方法は、

10

(a) 記録されたセッションにおける packets によって格納された入力のタイプを識別するステップであって、該記録されたセッションは、アプリケーションプログラムによって生成されたディスプレイデータを表す複数の packets を含むステップと、

(b) 該入力のタイプに応答する該 packets にマーカー付けするステップと、

(c) プレイバックデータ構造において、該マーカー付けすることに応答する該 packets のコンテンツをレンダリングする行先を格納するステップと

を包含する、方法。

【請求項 22】

前記ステップ (a) は、前記入力のタイプを、入力デバイスからの入力として識別することをさらに包含する、請求項 21 に記載の方法。

20

【請求項 23】

前記ステップ (a) は、前記入力のタイプを、キーボード入力として識別することをさらに包含する、請求項 21 に記載の方法。

【請求項 24】

前記ステップ (a) は、前記入力のタイプを、コマンドとして識別することをさらに包含する、請求項 21 に記載の方法。

【請求項 25】

前記ステップ (b) は、ポリシーに응答する前記 packets にマーカー付けすることをさらに包含する、請求項 21 に記載の方法。

【請求項 26】

30

前記ステップ (c) は、プレイバックデータ構造において、認知が可能な仕方で、バッファに前記マーカー付けされた packets のコンテンツをレンダリングする命令を格納することをさらに包含する、請求項 21 に記載の方法。

【請求項 27】

前記ステップ (c) は、前記プレイバックデータ構造において、バッファに前記マーカー付けされた packets のコンテンツをレンダリングする命令を格納することをさらに包含する、請求項 21 に記載の方法。

【請求項 28】

記録されたセッションをレンダリングする際に、不活性期間を消去する方法であって、該方法は、

40

(a) 記録されたセッションにおける第 1 の packets を識別するステップであって、該記録されたセッションは、複数の packets を含み、ディスプレイデータを表す、ステップと、

(b) 該記録されたセッションにおいて第 2 の packets を識別するステップであって、該第 2 の packets は、直ちに該第 1 の packets に先行する、ステップと、

(c) 該第 1 の packets と該第 2 の packets との間の第 1 の時間間隔を決定するステップと、

(d) 該第 1 の時間間隔が閾値を超えることを決定する、ステップと、

(e) 該第 1 の packets と該第 2 の packets との間の第 2 の時間間隔を用いて該記録されたセッションのコンテンツをレンダリングするステップであって、該第 2 の時間間隔は

50

、該第 1 の時間間隔より短い時間間隔を含むステップ
を包含する、方法。

【請求項 29】

前記ステップ (e) は、プレイバックデータ構造において、前記第 1 のパケットと前記第 2 のパケットとの間の前記第 2 の時間間隔を用いて前記記録されたセッションのコンテンツをレンダリングするための指示を格納することをさらに包含する、請求項 28 に記載の方法。

【請求項 30】

前記ステップ (e) は、前記記録されたセッションにおいて第 2 のパケットを識別し、該第 2 のパケットは、直ちに前記第 1 のパケットに続くことをさらに包含する、請求項 28 に記載の方法。

【請求項 31】

記録されたセッションをレンダリングする際に、意味のない活動を含む期間を消去する方法であって、該方法は、

(a) 記録されたセッションにおいて、マーカー付けされたパケットと、最も近い以前にマーカー付けされたパケットとの間の第 1 の時間間隔を決定するステップであって、該記録されたセッションは、複数のパケットを含み、ディスプレイデータを表すステップと、

(b) 該第 1 の時間間隔が、閾値を超えることを決定するステップと、

(c) 該マーカー付けされたパケットと、該最も近い以前にマーカー付けされたパケットとの間の第 2 の時間間隔を用いて該記録されたセッションのコンテンツをレンダリングするステップであって、該第 2 の時間間隔は、該第 1 の時間間隔より短い時間間隔を含むステップと

を包含する、方法。

【請求項 32】

前記ステップ (c) は、前記マーカー付けされたパケットと以前のパケットとの間の第 2 の時間間隔を用いて前記記録されたセッションのコンテンツをレンダリングすることをさらに包含し、該第 2 の時間間隔は、前記第 1 の時間間隔よりも短い時間間隔を含む、請求項 31 に記載の方法。

【請求項 33】

前記ステップ (c) は、前記マーカー付けされたパケットと、該マーカー付けされたパケットに続くパケットとの間の第 2 の時間間隔を用いて前記記録されたセッションのコンテンツをレンダリングすることをさらに包含し、該第 2 の時間間隔は、前記第 1 の時間間隔よりも短い時間間隔を含む、請求項 31 に記載の方法。

【請求項 34】

記録されたセッションをレンダリングする際に、グラフィックの更新を消去する方法であって、該方法は、

(a) アプリケーションプログラムによって生成されたディスプレイデータを表す記録されたセッションにおいてパケットを識別するステップであって、該パケットは、第 1 のグラフィックの更新を含むステップと、

(b) 該第 1 のグラフィックの更新によって影響を及ぼされたスクリーン領域を決定するステップと、

(c) 該第 1 のグラフィックの更新の後の該スクリーン領域の状態の指示と該スクリーン領域のロケーションとを格納するステップと、

(d) 時間間隔内に該スクリーン領域に影響を及ぼす第 2 のグラフィックの更新を識別するステップと、

(e) プレイバックデータ構造において、該スクリーン領域に影響を及ぼす該第 2 のグラフィックの更新を含む第 2 のパケットをレンダリングする行先を指示するステップであって、該第 2 のグラフィックの更新の後の該スクリーン領域の状態が、該第 1 のグラフィックの更新の後の該スクリーン領域の該状態と異なるかどうかに応答するステップと、

10

20

30

40

50

(f) プレイバックデータ構造において、該スクリーン領域に影響を及ぼす該第2のグラフィックの更新を含む該第2のパケットに関連付いた、レンダリングする時間間隔を指示するステップであって、該第2のグラフィックの更新の後の該スクリーン領域の状態が、該第1のグラフィックの更新の後の該スクリーン領域の状態と異なるかどうかに応答するステップと

を包含する、方法。

【請求項35】

前記ステップ(b)は、複数のパケット内に含まれたグラフィックの更新を識別することをさらに包含する、請求項34に記載の方法。

【請求項36】

前記ステップ(b)は、前記識別された複数のパケットにおける各パケットによって格納された前記グラフィックの更新に応答して、影響を及ぼされた前記スクリーン領域を決定することをさらに包含する、請求項34に記載の方法。

【請求項37】

前記ステップ(c)は、前記グラフィックの更新によって影響を及ぼされた前記スクリーン領域のコピーを格納することをさらに包含する、請求項34に記載の方法。

【請求項38】

前記ステップ(d)は、ヒューマンスケールの循環期間の上限を近似するように定義される時間間隔をさらに包含する、請求項34に記載の方法。

【請求項39】

前記ステップ(e)は、前記バックグラウンドプロトコルエンジンによって、プレイバックデータ構造において、前記パケットのコンテンツは、認識が可能な仕方において、バッファにレンダリングすることを示すことをさらに包含する、請求項34に記載の方法。

【請求項40】

前記ステップ(e)は、前記バックグラウンドプロトコルエンジンによって、プレイバックデータ構造において、前記パケットの前記コンテンツが、バッファにレンダリングすることを示すことをさらに包含する、請求項34に記載の方法。

【請求項41】

記録されたセッションをレンダリングする際に、相互作用シーケンスを消去する方法であって、該方法は、

(a) ポリシーに応答して、少なくとも一つのパケットの相互作用シーケンスの開始を識別するステップと、

(b) プレイバックデータ構造において、相互作用シーケンスは、バッファにレンダリングすべきであることを示す、ステップと、

(c) 相互作用シーケンスの終了を識別するステップと、

(d) 相互作用シーケンスの識別された開始に先行するパケットと相互作用シーケンスの識別された終了に続くパケットとの間に第1の時間間隔を識別するステップと、

(e) プレイバックデータ構造において、第1の時間間隔より短い、レンダリングする第2の時間間隔を示すステップと

を包含する、方法。

【請求項42】

相互作用シーケンスは、ログオンシーケンスを含む、請求項41に記載の方法。

【請求項43】

相互作用シーケンスは、ログオフシーケンスを含む、請求項41に記載の方法。

【請求項44】

ポリシーは、相互作用シーケンスを定義する、請求項41に記載の方法。

【請求項45】

前記ステップ(a)は、視覚的なマーカーを識別することによって相互作用シーケンスの開始を識別することをさらに包含する、請求項41に記載の方法。

【請求項46】

10

20

30

40

50

前記ステップ (a) は、相互作用シーケンスの開始時間を決定することによって該相互作用シーケンスの前記開始を識別することをさらに包含する、請求項 4 1 に記載の方法。

【請求項 4 7】

前記ステップ (a) は、入力フォーカスを有するウィンドウを識別することによって前記相互作用シーケンスの前記開始を識別することをさらに包含する、請求項 4 1 に記載の方法。

【請求項 4 8】

前記ステップ (c) は、視覚的なマーカーを識別することによって前記相互作用シーケンスの前記終了を識別することをさらに包含する、請求項 4 1 に記載の方法。

【請求項 4 9】

前記ステップ (c) は、相互作用シーケンスの終了時間を決定することによって該相互作用シーケンスの前記終了を識別することをさらに包含する、請求項 4 1 に記載の方法。

【請求項 5 0】

前記ステップ (c) は、入力フォーカスを有するウィンドウを識別することによって前記相互作用シーケンスの前記終了を識別することをさらに包含する、請求項 4 1 に記載の方法。

【請求項 5 1】

記録されたコンピュータセッションのレンダリングの際の自動的なタイムワーブプレイバックの方法であって、該方法は、

(a) バックグラウンドプロトコルエンジンによって、記録されたセッションを受信するステップであって、該記録されたセッションは、複数のパケットを含み、ディスプレイデータを表すステップと、

(b) 該バックグラウンドプロトコルエンジンによって、該記録されたセッションにおける該複数のパケットの少なくとも一部によって表された複雑さの程度を決定するステップと、

(c) 該バックグラウンドプロトコルエンジンによって、該記録されたセッションにおける該複数のパケットの該少なくとも一部の間の時間の間隔を識別するステップと、

(d) 該バックグラウンドプロトコルエンジンによって、該記録されたセッションにおける該複数のパケットの該少なくとも一部によって表された複雑さの程度に回答して、該時間の間隔を修正するステップと、

(e) 該バックグラウンドプロトコルエンジンによって、プレイバックデータ構造において、該修正された時間の間隔を格納するステップと、

(f) フォアグラウンドプロトコルエンジンによって、該プレイバックデータ構造に回答する、該記録されたストリームをレンダリングするステップとを包含する、方法。

【請求項 5 2】

前記ステップ (b) は、キーボード入力におけるタイピングの適切なシーケンスを識別することをさらに包含する、請求項 5 1 に記載の方法。

【請求項 5 3】

前記ステップ (b) は、関係したキーの少なくとも一つのタイプを検査することをさらに包含する、請求項 5 2 に記載の方法。

【請求項 5 4】

前記ステップ (b) は、キーボード入力におけるタイピングの適切なシーケンスの発見的な近似を完成させるためにレンダリングされる少なくとも一つのグリフのシーケンスを検査することをさらに包含する、請求項 5 1 に記載の方法。

【請求項 5 5】

前記ステップ (b) は、マウス入力のレートを識別することをさらに包含する、請求項 5 1 に記載の方法。

【請求項 5 6】

前記ステップ (b) は、グラフィックの更新の複雑さの発見的な近似を識別することを

10

20

30

40

50

さらに包含する、請求項 5 1 に記載の方法。

【請求項 5 7】

前記ステップ (b) は、前記記録されたセッションにおいて前記複数のパケットの前記少なくとも一部における第 1 のマーカを識別することをさらに包含し、該マーカは、該記録されたセッションにおいて該複数のパケットの該少なくとも一部における最初のパケットを示す、請求項 5 1 に記載の方法。

【請求項 5 8】

前記ステップ (d) は、前記第 1 のマーカに応答して、前記時間の間隔を修正することをさらに包含する、請求項 5 7 に記載の方法。

【請求項 5 9】

前記ステップ (d) は、前記記録されたセッションにおける前記複数のパケットの前記少なくとも一部における、第 2 のパケットにおける第 2 のマーカを識別することをさらに包含し、該第 2 のマーカは、該記録されたセッションにおける該複数のパケットの該少なくとも一部における最後のパケットを示し、該第 1 のパケットと該第 2 のパケットとの間の該時間の間隔に
10 応答して、該時間の間隔を修正する、請求 5 7 に記載の方法。

【請求項 6 0】

前記ステップ (d) は、前記時間の間隔を減少させることをさらに包含する、請求項 5 1 に記載の方法。

【請求項 6 1】

前記ステップ (d) は、前記時間の間隔を増加させることをさらに包含する、請求項 5 1 に記載の方法。
20

【請求項 6 2】

前記ステップ (d) は、要求に応答して、前記時間の間隔を修正することをさらに包含する、請求項 5 1 に記載の方法。

【請求項 6 3】

記録されたコンピュータセッションをレンダリングする際の自動的なタイムワーブプレイバックの方法であって、該方法は、

(a) 複数のパケットを含み、ディスプレイデータを表す記録されたセッションを受信するステップと、

(b) フォーカスを有するウィンドウを表すコンテンツを有する第 1 のパケットを識別するステップであって、該ウィンドウは、アプリケーションを示すステップと、
30

(c) 第 1 のパケットのコンテンツのレンダリングに先行して、コンテンツがレンダリングする第 2 のパケットと、該第 1 のパケットのコンテンツのレンダリングの後に、コンテンツがレンダリングする第 3 のパケットとの間で、時間間隔を識別するステップと、

(d) 該示されたアプリケーションに応答して該識別された時間間隔を修正するステップと、

(e) 該修正に応答して、該記録されたストリームにおける少なくとも一つのパケットをレンダリングするステップと

を包含する、方法。
40

【請求項 6 4】

前記ステップ (d) は、前記識別された時間間隔を増加させることをさらに包含する、請求項 6 3 に記載の方法。

【請求項 6 5】

前記ステップ (d) は、前記識別された時間間隔を減少させることをさらに包含する、請求項 6 3 に記載の方法。

【請求項 6 6】

前記ステップ (e) は、前記記録されたストリームにおける前記少なくとも一つのパケットをバッファにレンダリングすることをさらに包含する、請求項 6 3 に記載の方法。

【請求項 6 7】

前記ステップ (e) は、前記記録されたストリームにおける前記少なくとも一つのパケ
50

ットを、バッファに、認知が可能な仕方において、レンダリングすることをさらに包含する、請求項 63 に記載の方法。

【請求項 68】

記録されたコンピュータセッションをレンダリングする際の自動的なタイムワーププレイバックのシステムであって、該システムは、

プロトコルエンジンであって、記録されたストリームを受信することに応答して、プレイバックデータ構造を生成し、該記録されたストリームは、複数のパケットを含み、該プロトコルエンジンは、該生成されたプレイバックデータ構造に応答して、該記録されたストリームにおける少なくとも一つのパケットをレンダリングする、プロトコルエンジンを備えた、システム。

10

【請求項 69】

前記プロトコルエンジンは、前記プレイバックデータ構造において、前記記録されたストリームをレンダリングするための少なくとも一つの命令を格納することをさらに包含する、請求項 68 に記載のシステム。

【請求項 70】

前記プロトコルエンジンは、前記プレイバックデータ構造においてメタデータを格納することをさらに包含する、請求項 69 に記載のシステム。

【請求項 71】

前記プロトコルエンジンは、認知が可能な仕方において、バッファに、前記記録されたセッションにおける前記少なくとも一つのパケットをレンダリングする、請求項 68 に記載のシステム。

20

【請求項 72】

前記プロトコルエンジンは、前記記録されたセッションにおける前記少なくとも一つのパケットを、バッファにレンダリングする、請求項 68 に記載のシステム。

【請求項 73】

第2のプロトコルエンジンは、前記生成されたプレイバックデータ構造に応答して、前記記録されたストリームにおける前記少なくとも一つのパケットをレンダリングする、請求項 68 に記載のシステム。

【発明の詳細な説明】

【技術分野】

30

【0001】

本発明は、プレイバック命令を生成する方法およびシステムに関連し、特に、記録されたコンピュータセッションをレンダリングする際に、自動タイムワーププレイバックを含む、プレイバックに対する命令を生成することに関連する。

【背景技術】

【0002】

遠隔提示プロトコルである、Citrix Systems, Inc. of Fort Lauderdale, Florida製のICAプロトコル、the X.org Foundation製のXプロトコル、AT&T Corp.製のVirtual Network Computingプロトコル、またはMicrosoft Corporation of Redmond, Washington製のRDPプロトコルなどは、本質的に処理状態を把握できる。記録された提示プロトコルデータのストリームにおいて特定のポイントを調べるために、該ストリームのプレイバックは、ストリームの最初から行わなければならない、所望のポイントが現われるまで、連続して再生されなくてはならない。

40

【0003】

セッションを記録する多くの従来の方法は、スクリーンナップショットを定期的に撮ることによって動作し、これらの一部は、セッション活動から少しずつ収集された潜在的な関心の指示に応答して、ナップショットの頻度を増加させる。画像のシーケンスは、スライドショーとして見られ得るか、画像コレクション管理ツールを用いて見られ得る。

50

他の方法は、フレームベースのデジタルビデオフォーマット、例えばM P GまたはA V Iに記録し得、これらは、適切なメディアプレーヤー、例えばW i n d o w s（登録商標）M e d i a P l a y e rまたはQ u i c k T i m e p l a y e rなどで見られる。多くの従来の方法は、特定のウィンドウまたはアプリケーションを伴う相互作用を示すセクションを消去することによってレビュータイムを低減するという性能を欠く。

【0004】

一部の従来の方法は、リアルタイムレートのマルチプルで記録されたセッションのプレイバックを可能にする。ユーザは、それらのマルチプルのうちのいずれか一つにおいて再生することを選択し得、プレイバックの間にスピード倍率器を変更し得る。しかし、ユーザは、これから何がレンダリングされるかが分らないので、彼らは、関心の低いセクションの間に速度を速める傾向にあり、より高い関心のセクションが始まるときには、詳細を逃してしまう。さらに、リアルタイムよりも何倍も速い速度であっても、無意味なユーザ活動の長いセクションをレビューしているときは、主観的に遅いものである。

10

【0005】

多くの従来システムは、スナップショット生成を最小限にし、ストリームトラバーサル速度を増加させることによって、プレイバックを最適化するように試みてきた。一部の場合において、これらのシステムの一部は、重要な活動が検出されるまで、より少ないスクリーンスナップショットを行い、次いでスナップショットの頻度を増加させる。これらのシステムの一部は、スナップショットの間の入力および状態の変化がキャプチャされないときに、状態をなくしてしまうという欠点を被り得る。

20

【発明の開示】

【課題を解決するための手段】

【0006】

本発明は、遠隔提示プロトコルである、C i t r i x S y s t e m s、I n c . o f F t . L a u d e r d a l e、F l o r i d a製のI C Aプロトコル、t h e X . o r g F o u n d a t i o n製のXプロトコル、A T & T C o r p .製のV i r t u a l N e t w o r k C o m p u t i n gプロトコル、またはM i c r o s o f t C o r p o r a t i o n o f R e d m o n d、W a s h i n g t o n製のR D Pプロトコルなどの記録およびプレイバックの方法を提供する。本発明は、重要なことは何も起こらないことを、アルゴリズムによって決定され得るレコーディングの1つ以上のセクションをプレイバックから除去することによって、セッションレコーディングを手動でレビューするのに費やされる時間を減少させる。本発明は、有向再生、すなわち記録されたセッションの代替的なレンダリングを提供する。本発明は、オフスクリーンレンダリングオペレーションを高めることによって、どのように有向再生を行うかを記述するプレイバックデータ構造を生成し、そのプレイバックデータ構造を用いることによってオンスクリーンレンダリングプロセスを制御する。ユーザ選択されたリアルタイムのマルチプルのみを提供するのではなく、本発明は、査閲者によって付加的に変調された、自動的に変更される文脈依存再生レートを提供する。本発明は、査閲者に情報を、彼らの理解力のレートに近似する自動的に選ばれたレートにおいて提供する。

30

【0007】

一局面において、本発明は、記録されたコンピュータセッションのプレイバックに対するプレイバック命令を生成する方法に関連する。第1のデバイス上で実行するプロトコルエンジンは、第2のデバイス上で実行されるアプリケーションプログラムによって生成されたディスプレイデータを表す複数のパケットを含む、記録されたストリームを受け取る。プロトコルエンジンは、記録されたストリームにおけるパケットに対して、人間が感知することが可能な態様でパケットのコンテンツをレンダリングすることを決定する。プロトコルエンジンは、該決定をプレイバックデータ構造に格納する。

40

【0008】

別の実施形態において、本発明は、記録されたコンピュータセッションのプレイバックの方法に関連する。第1のデバイス上で実行するバックグラウンドプロトコルエンジンは

50

、第2のデバイス上で実行するアプリケーションプログラムによって生成されたディスプレイデータを表す複数のパケットを含む記録されたストリームを受信する。バックグラウンドプロトコルエンジンは、人間が感知することが可能な態様でパケットをレンダリングするように決定する。バックグラウンドプロトコルエンジンは、プレイバックデータ構造において該決定を格納する。フォアグラウンドプロトコルエンジンは、記録されたストリームから少なくとも一つのパケットを検索する。フォアグラウンドプロトコルエンジンは、プレイバックデータ構造にアクセスする。フォアグラウンドプロトコルエンジンは、プレイバックデータ構造に応答して、少なくとも一つのパケットをレンダリングする。

【0009】

さらに別の局面において、本発明は、記録されたセッションのレンダリングのシステムに関連する。第1のデバイス上で実行するバックグラウンドプロトコルエンジンは、記録されたストリームの受信に応答して、プレイバックデータ構造を生成し、該記録されたストリームは、第2のデバイス上で実行されたアプリケーションプログラムによって生成されたディスプレイデータを表す。フォアグラウンドプロトコルエンジンは、記録されたストリームを受信し、バックグラウンドプロトコルエンジンによって生成されたプレイバックデータ構造に応答して、記録されたストリームをレンダリングする。

【0010】

一局面において、本発明は、記録されたセッションをレンダリングするプレイバック命令を生成する方法に関連する。記録されたセッションにおいてパケットによって格納される入力のタイプは、識別され、該記録されたセッションは、アプリケーションプログラムによって生成されたディスプレイデータを表す複数のパケットを含む。パケットは、入力のタイプに応答してマーカー付けされる。パケットのコンテンツをレンダリングする行先は、マーカー付けに応答して、プレイバックデータ構造に格納される。

【0011】

一局面において、本発明は、記録されたセッションをレンダリングする際に、不活動の期間を消去する方法に関連する。記録されたセッションにおける第1のパケットは、識別され、該記録されたセッションは、複数のパケットを含み、ディスプレイデータを表す。記録されたセッションにおける第2のパケットは、識別され、該第2のパケットは、直ちに第1のパケットに先行する。第1のパケットと第2のパケットとの間の第1の時間間隔は、決定される。第1の時間間隔は、閾値を超えるという決定がなされる。第1のパケットと第2のパケットとの間の第2の時間間隔を有する記録されたセッションのコンテンツが識別され、該第2の時間間隔は、第1の時間間隔より短い時間間隔を含む。一実施形態において、第2のパケットは、第1のパケットに直ちに続く。

【0012】

別の局面において、本発明は、記録されたセッションをレンダリングする際に、不活動の期間を消去する方法に関連する。第1の時間間隔は、記録されたセッションにおけるマーク付けされたパケットと、最も近い以前のマーク付けされたパケットとの間に決定され、該記録されたセッションは、複数のパケットを含み、ディスプレイデータを表す。第1の時間間隔は、閾値を超えるということが決定される。記録されたセッションのコンテンツは、マーク付けされたパケットと、最も近い以前のマーク付けされたパケットとの間の第2の時間間隔を用いてレンダリングされ、該第2の時間間隔は、第1の時間間隔より短い時間間隔を含む。

【0013】

さらに別の局面において、本発明は、記録されたセッションをレンダリングする際に、グラフィックの更新を消去する方法に関連する。記録されたセッションにおける、アプリケーションプログラムによって生成されたディスプレイデータを表すパケットが、識別され、該パケットは、第1のグラフィックの更新を含む。第1のグラフィックの更新によって影響を及ぼされたスクリーン領域が、決定される。指示は、第1のグラフィックの更新の後のスクリーン領域の状態およびスクリーン領域のロケーションが格納される。時間間隔内にスクリーン領域に影響を及ぼす第2のグラフィックの更新が、識別される。スクリ

ーン領域に影響を及ぼす第2のグラフィックの更新を含む、第2の packets をレンダリングする行先は、プレイバックデータ構造に示され、第2のグラフィックの更新の後のスクリーン領域の状態が、第1のグラフィックの更新の後のスクリーン領域の状態と変わるかどうかに応答する。スクリーン領域に影響を及ぼす第2のグラフィックの更新を含む第2の packets に関連した、レンダリングする時間間隔は、プレイバックデータ構造において示され、第2のグラフィックの更新の後のスクリーン領域の状態が、第1のグラフィックの更新の後のスクリーン領域の状態と変わるかどうかに応答する。

【0014】

さらに別の局面において、本発明は、記録されたセッションをレンダリングする際に、相互作用シーケンスを消去する方法に関連する。ポリシーに応答して、少なくとも一つの packets の相互作用シーケンスの開始が識別される。相互作用シーケンスは、バッファにレンダリングされるべきであるということが、プレイバックデータ構造に示される。相互作用シーケンスの終了が、識別される。第1の時間間隔は、相互作用シーケンスの識別された開始に先立つ packets と相互作用シーケンスの識別された終了に続く packets との間に識別される。第1の時間間隔より短いレンダリングする第2の時間間隔が、プレイバックデータ構造において示される。

【0015】

一局面において、本発明は、記録されたコンピュータセッションをレンダリングする際の、自動的なタイムワーププレイバックの方法に関連する。バックグラウンドプロトコルエンジンは、記録されたセッションを受信し、該記録されたセッションは、複数の packets を含み、ディスプレイデータを表す。バックグラウンドプロトコルエンジンは、記録されたセッションにおける複数の packets の少なくとも一部によって表される複雑さの程度を決定する。バックグラウンドプロトコルエンジンは、記録されたセッションにおける複数の packets のうちの少なくとも一部の間の時間の間隔を識別する。バックグラウンドプロトコルエンジンは、記録されたセッションにおける複数の packets の少なくとも一部によって表される複雑さの程度に応答する、識別された時間の間隔を修正する。バックグラウンドプロトコルエンジンは、プレイバックデータ構造において、修正された時間の間隔を格納する。フォアグラウンドプロトコルエンジンは、プレイバックデータ構造に応答して、記録されたストリームをレンダリングする。

【0016】

別の局面において、本発明は、記録されたコンピュータセッションの際の、自動的なタイムワーププレイバックの別の方法に関連する。複数の packets を含み、ディスプレイデータを表す記録されたセッションが、受信される。フォーカスを有するウィンドウを表しているコンテンツを有する第1の packets が、識別され、該ウィンドウは、アプリケーションを示す。第1の packets のコンテンツのレンダリングに先立ち、コンテンツがレンダリングする第2の packets と、第1の packets のコンテンツのレンダリングの後に、コンテンツがレンダリングする第3の packets との間で、時間間隔は識別される。時間間隔は、示されたアプリケーションに応答して修正される。記録されたストリーム内の少なくとも一つの packets は、修正に応答してレンダリングされる。

【0017】

別の局面において、本発明は、記録されたコンピュータセッションをレンダリングする際の、自動的なタイムワーププレイバックのシステムに関連する。プロトコルエンジンは、記録されたストリームに応答してプレイバックデータ構造を生成し、該記録されたストリームは、複数の packets を含み、該プロトコルエンジンは、生成されたプレイバックデータ構造に応答して、記録されたストリームにおける少なくとも一つの packets をレンダリングする。

【発明を実施するための最良の形態】

【0018】

本発明のこれらおよび他の局面は、本明細書中の詳細な記述および添付される図面によって直ちに明確になる。これらは、本発明を説明するものであり、限定する意図はない。

【0019】

ここで図1Aを参照して、簡潔に概略すると、本発明が用いられ得るクライアント・サーバシステムの一実施形態が図示される。第1のコンピューティングデバイス100'（一般に100）は、通信ネットワーク180を介して第2のコンピューティングデバイス140'（一般に140）と通信する。第1のデバイス100が介して第2のデバイス140と通信するネットワーク180のトポロジは、バス、スターまたはリングのトポロジであり得る。ネットワーク180は、ローカルエリアネットワーク（LAN）、メトロポリタンエリアネットワーク（MAN）、または、例えばインターネットのようなワイドエリアネットワーク（WAN）であり得る。2つの第1のコンピューティングデバイス100、100'および2つの第2のコンピューティングデバイス140、140'のものが、図1Aに図示されるが、他の実施形態は、ネットワーク180に接続される複数のそのようなデバイスを含む。

10

【0020】

第1のデバイス100および第2のデバイス140は、様々な接続を介してネットワーク180に接続し得、接続は、標準的な電話線と、LANまたはWANリンク（例えば、T1、T3、56kb、X.25）と、ブロードバンド接続（ISDN、フレームリレー、ATM）と、ワイアレス接続とを含む。接続は、様々な通信プロトコルを用いて確立され得る（例えば、TCP/IP、IPX、SPX、NetBIOS、NetBEUI、SMB、Ethernet（登録商標）、ARCNET、Fiber Distributed Data Interface（FDDI）、RS232、IEEE802.11、IEEE802.11a、IEEE802.11b、IEEE802.11gおよび直接的で非同期式の接続）。

20

【0021】

第1のデバイス100は、該デバイスに代わって、1つ以上の第2のコンピューティングデバイス140によって実行されるアプリケーションからの出力を受信および表示する能力があり、本明細書に開示されるプロトコルに従って動作する能力がある任意のデバイスであり得る。第1のデバイス100は、パーソナルコンピュータ、ウィンドウズ（登録商標）ベースのターミナル、ネットワークコンピュータ、情報アプライアンス、X-デバイス、ワークステーション、ミニコンピュータ、携帯情報端末、または携帯電話であり得る。

30

【0022】

同様に、第2のコンピューティングデバイス140は、任意のコンピューティングデバイスであり得、実行するアプリケーションに対するユーザ入力を第1のコンピューティングデバイス100から受信する能力と、第1デバイス100の代わりにアプリケーションプログラムを実行する能力と、本明細書に開示されるプロトコルを用いて第1のコンピューティングデバイス100と相互作用する能力がある。第2のコンピューティングデバイス140は、シングルサーバシステムとして論理的に作用するサーバデバイスのグループとして提供され得、本明細書中においてサーバファームと呼ばれる。一実施形態において、第2のコンピューティングデバイス140は、マルチユーザサーバシステムであり、該システムは、もう1つの第1のデバイス100から複数の同時にアクティブな接続をサポートする。

40

【0023】

図1Bおよび図1Cは、第1のコンピューティングデバイス100と第2のコンピューティングデバイス140として有用な、一般的なコンピュータ100のブロック図を図示する。図1Bおよび図1Cに示されるように、中央処理装置102とメインメモリユニット104とを含む。各コンピュータ100はまた、他のオプションの要素を含み得、例えば1つ以上の入力/出力デバイス130aおよび130b（通常は、参照番号130を用いて参照される）と、中央処理装置102と通信するキャッシュメモリ145とである。

【0024】

中央処理装置102は、メインメモリユニット104から取り込まれた（fetch）

50

d) 命令に応答し処理する任意の論理回路網である。多くの実施形態において、中央処理装置は、マイクロプロセッサユニットによって提供され、該マイクロプロセッサユニットは、例えば、Intel Corporation of Mountain View, Californiaによって全て製造された、8088プロセッサ、80286プロセッサ、80386プロセッサ、80486プロセッサ、Pentium(登録商標)プロセッサ、Pentium(登録商標)プロセッサ、Pentium(登録商標)IIプロセッサ、Celeronプロセッサ、またはXeonプロセッサ、Motorola Corporation of Schaumburg, Illinoisによって全て製造された、68000プロセッサ、68010プロセッサ、68020プロセッサ、68030プロセッサ、68040プロセッサ、Power PC601プロセッサ、Power PC604プロセッサ、Power PC604eプロセッサ、MPC603eプロセッサ、MPC603eiプロセッサ、MPC603evプロセッサ、MPC603rプロセッサ、MPC603pプロセッサ、MPC740プロセッサ、MPC745プロセッサ、MPC750プロセッサ、MPC755プロセッサ、MPC7400プロセッサ、MPC7410プロセッサ、MPC7441プロセッサ、MPC7445プロセッサ、MPC7447プロセッサ、MPC7450プロセッサ、MPC7451プロセッサ、MPC7455プロセッサ、MPC7457プロセッサ、Transmeta Corporation of Santa Clara, Californiaによって製造された、Crusoe TM5800プロセッサ、Crusoe TM5600プロセッサ、Crusoe TM5500プロセッサ、Crusoe TM5400プロセッサ、Efficeon TM8600プロセッサ、Efficeon TM8300プロセッサ、またはEfficeon TM8620プロセッサ、International Business Machines of White Plains, New Yorkによって全て製造された、RS/6000プロセッサ、RS64プロセッサ、RS64IIプロセッサ、P2SCプロセッサ、POWER3プロセッサ、RS64IIプロセッサ、POWER3-IIプロセッサ、RS64IVプロセッサ、POWER4プロセッサ、POWER4+プロセッサ、POWER5プロセッサ、またはPOWER6プロセッサ、またはAdvanced Micro Devices of Sunnyvale, Californiaによって製造された、AMD Opteronプロセッサ、AMD Athlon 64 FXプロセッサ、AMD Athlonプロセッサ、またはAMD Duronプロセッサである。

【0025】

メインメモリユニット104は、データを格納する能力と、任意の格納ロケーションがマイクロプロセッサ102によって直接アクセスされることを可能にする能力とを有する1つ以上のメモリチップであり得、例えば、スタティックランダムアクセスメモリ(SRAM)、バースト(Burst)SRAMまたはSynchBurst SRAM(BSRAM)、ダイナミックランダムアクセスメモリ(DRAM)、ファストページモードDRAM(FPM DRAM)、エンハンスドDRAM(EDRAM)、エクステンデッドデータ出力RAM(EDO RAM)、エクステンデッドデータ出力DRAM(EDO DRAM)、バーストエクステンデッドデータ出力DRAM(BEDO DRAM)、エンハンスドDRAM(EDRAM)、シンクロノスDRAM(SDRAM)、JEDEC SRAM、PC100 SDRAM、ダブルデータレートSDRAM(DDR SDRAM)、エンハンスドSDRAM(ESDRAM)、SyncLinc DRAM(SLDRAM)、ディレクトランバスDRAM(DRDRAM)、またはFerroelectric RAM(FRAM)などである。図1Bに示される実施形態において、プロセッサ102は、システムバス120を介してメインメモリ104と通信する(以下に詳細に記述される)。図1Cは、プロセッサが、メモリポートを介してメインメモリ104と直接的に通信するコンピュータシステム100の実施形態を図示する。例えば、図1Cにおいて、メインメモリ104はDRDRAMであり得る。

【0026】

10

20

30

40

50

図1Bおよび図1Cは、メインプロセッサ102が、時に「バックサイド」バスと呼ばれる二次的なバスを介してキャッシュメモリ145と直接的に通信する実施形態を図示する。他の実施形態において、メインプロセッサ102は、システムバス120を用いてキャッシュメモリ145と通信する。キャッシュメモリ145は、通常はメインメモリ104よりも速い応答時間を有し、通常はSRAM、BSRAMまたはEDRAMによって提供される。

【0027】

図1Bに示される実施形態において、プロセッサ102は、ローカルシステムバス120を介して様々なI/Oデバイス130と通信する。様々なバスは、中央処理装置102をI/Oデバイス130に接続するために用いられ得、該バスは、VESA VLバス、ISAバス、EISAバス、Micro Channel Architecture (MCA) バス、PCIバス、PCI-Xバス、PCI-Expressバス、またはNuBusを含む。I/Oデバイスがビデオディスプレイである実施形態に対して、プロセッサ102は、ディスプレイと通信するためにアドバンスドグラフィックポート (AGP) を用い得る。図1Cは、メインプロセッサ102が、ハイパートランスポート、ラビッドI/O、またはInfiniBandを用いてI/Oデバイス130bと直接的に通信する、コンピュータシステム100の実施形態を図示する。図1Cはまた、ローカルバスと直接的な通信とが交じり合った実施形態を図示し、プロセッサ102は、ローカルインターコネクトバスを用いてI/Oデバイス130aと通信し、その一方でI/Oデバイス130bと直接的に通信する。

10

20

【0028】

幅広い種類のI/Oデバイス130が、コンピュータシステム100において存在し得る。入力デバイスは、キーボードと、マウスと、トラックパッドと、トラックボールと、マイクと、製図 (drawing) タブレットとを含む。出力デバイスは、ビデオディスプレイと、スピーカと、インクジェットプリンタと、レーザープリンタと、昇華型プリンタとを含む。I/Oデバイスはまた、コンピュータシステム100に対して大容量のストレージを提供し得、例えば、ハードディスクドライブと、3.5インチ、5.25インチ、またはZIPディスクなどのフロッピー (登録商標) ディスクを受けるためのフロッピー (登録商標) ディスクドライブと、CD-ROMドライブと、CD-R/RWドライブと、DVD-ROMドライブと、DVD-Rドライブと、DVD-RWドライブと、様々なフォーマットのテープドライブと、Twintech Industry, Inc. of Los Alamitos, Californiaによって製造されたUSBフラッシュドライブなどの種類のデバイスUSB格納デバイスとである。

30

【0029】

さらなる実施形態において、I/Oデバイス130は、システムバス120と外部通信バスとの間のブリッジであり得、該外部通信バスは、例えばUSBバス、Appleデスクトップバス、RS-232シリアル接続、SCSIバス、FireWireバス、FireWire 800バス、Ethernet (登録商標) バス、Appleトークバス、ギガビットEthernet (登録商標) バス、非同期式のトランスファーモードバス、HIPPIバス、スーパーHIPPIバス、シリアルプラスバス、SCI/LAMPバス、Fibre Channelバス、またはシリアルアタッチト小型コンピュータシステムインターフェースバスである。

40

【0030】

図1Bおよび図1Cに図示される種類の汎用デスクトップコンピュータは、通常オペレーティングシステムの制御の下で動作し、該オペレーティングシステムは、タスクのスケジューリングとシステムリソースへのアクセスとを制御する。通常のオペレーティングシステムが含むものは、とりわけ、Microsoft Corp. of Redmond, Washingtonによって製造されたMICROSOFT WINDOWS (登録商標) と、Apple Computer of Cupertino, Californiaによって製造されたMac OSと、International Busi

50

ness Machines of Armonk New Yorkによって製造されたOS/2と、Caldera Corp. of Salt Lake City, Utahによって配給された自由に利用可能なオペレーティングシステムであるLinuxとである。

【0031】

他の実施形態において、第1のデバイス100または第2のデバイス140は、デバイスと調和する異なるプロセッサと、オペレーティングシステムと、入力デバイスとを有し得る。例えば、一実施形態において、第1のデバイス100は、Palm, Inc.によって製造された携帯情報端末のZire 71である。この実施形態において、Zire 71は、Texas Instruments, of Dallas, Texasによって製造されたOMAP 310プロセッサを用い、PalmOSオペレーティングシステムの制御の下で動作し、液晶表示スクリーンと、スタイラス入力デバイスと、5方向ナビゲーターデバイスとを含む。

10

【0032】

ここで図2を参照して、ブロック図が、ネットワーク200の実施形態を図示し、該ネットワーク200の中で本発明が実行され得、第1のデバイス202と、遠隔提示プロトコルサーバエンジン204と、レコーダ206と、プロトコルデータストリーム208と、レコーデッドプロトコルデータストリーム210と、第2のデバイス212と、遠隔提示プロトコルクライアントエンジン214と、ディスプレイ216と、格納要素218と、レコーデッドプロトコルデータストリーム220とを含む。簡潔に概略すると、レコーダ206は、プロトコルデータストリーム208をインターセプトする。レコーダ206は、プロトコルデータストリームから少なくとも1つのパケットをコピーし、該少なくとも1つのコピーされたパケットを用いてプロトコルデータストリーム210の記録を作成する。

20

【0033】

ここで図2を参照して、さらに詳細には、第1のデバイス202は、プロトコルデータストリーム208を第2のデバイス212に送信する。一実施形態において、第1のデバイス202は、遠隔提示プロトコルサーバエンジン204を用いて第2のデバイス212にプロトコルデータストリーム208を送信する。一部の実施形態において、第2のデバイス212は、遠隔提示プロトコルクライアントエンジン214を用いて第1のデバイス202からプロトコルデータストリーム208を受信する。一部の実施形態において、遠隔提示プロトコルは、小型軽量クライアントプロトコル、例えばCitrix Systems, Inc., of Ft. Lauderdale, Floridaによって製造されたICAプロトコル、X.org FoundationによるXプロトコル、AT&T Corp.のVirtual Network Computing protocol、またはMicrosoft Corporation of Redmond, Washingtonによって製造されたRDPプロトコルを含む。

30

【0034】

プロトコルデータストリーム208は、複数のパケットを含み、その少なくとも一部は、ディスプレイデータを表す。一部の実施形態において、プロトコルデータストリーム208は、記録されたセッションについての情報を含む。一実施形態において、プロトコルデータストリーム208は、メタデータを含む。別の実施形態において、プロトコルデータストリーム208は、記録されたセッションにおけるユーザについての情報を含む。さらに別の実施形態において、プロトコルデータストリーム208は、記録されたデータを生成するサーバについての情報を含む。さらなる別の実施形態において、プロトコルデータストリーム208は、タイムスタンプを含む。

40

【0035】

一実施形態において、プロトコルデータストリーム208は、複数のチャネルを含む。この実施形態において、チャネルは、データが送信されるピアトゥピアの接続を含む。別の実施形態において、プロトコルデータストリーム208は、複数の仮想のチャネルを含

50

む。この実施形態において、仮想のチャネルは、別のチャネルにラップ (w r a p p e d) されたチャネルである。第2のデバイス212は、プロトコルデータストリーム208を受信し、一部の実施形態において遠隔提示プロトコルクライアントエンジン214を用いてディスプレイデータを再生成する。プロトコルデータストリーム208を処理することは、第2のデバイス212がディスプレイ216を介してユーザにディスプレイを提示することを可能にする。第2のデバイス212は、ディスプレイデータを処理するために遠隔提示プロトコルクライアントエンジン214を用い得る。ディスプレイは、聴覚的、視覚的、触覚的、または嗅覚的な提示、またはこれらの組み合わせを含み、それらに限定はされない。

【0036】

10

レコーダ206は、第1のデバイス202から第2のデバイス212に送信されたプロトコルデータストリーム208をインターセプトする。一実施形態において、レコーダ206は、1つ以上のチャネルをインターセプトすることによってプロトコルデータストリーム208をインターセプトする。別の実施形態において、レコーダ206は、1つ以上の仮想のチャネルをインターセプトすることによってプロトコルデータストリーム208をインターセプトする。一部の実施形態において、レコーダ206は、第1のデバイス202が第2のデバイス212にプロトコルデータストリーム208を送信し得る1つ以上の仮想のチャネルをモニターする。レコーダ206は、プロトコルデータストリームから少なくとも1つのパケットをコピーする。一実施形態において、レコーダ206は、ポリシーに응答するプロトコルデータストリームの特定のパケットをコピーすることを決定する。一部の実施形態において、ポリシーは、パケット内に含まれるデータの種に基づいて、レコーダ206が記録するパケットを定義する。別の実施形態において、レコーダ206は、パケットがデータを含むかどうかという決定に基づいて、プロトコルデータストリームのパケットをコピーすることを決定する。これらの実施形態の一部において、レコーダ206は、空のパケットは記録せず、その一方で、他のこれらの実施形態において、レコーダ206は、空のパケットを記録する。一部の実施形態において、レコーダ206は、プロトコルデータストリーム208の中の全てのパケットを記録する。

20

【0037】

レコーダ206は、少なくとも1つのコピーされたパケットを用いて、記録されたプロトコルデータストリーム210を作成する。一実施形態において、レコーダ206は、情報を少なくとも1つのコピーされたパケットと関連付ける。一実施形態において、レコーダ206は、タイムスタンプを少なくとも1つのコピーされたパケットと関連付ける。別の実施形態において、レコーダ206は、データ長インジケータをパケットと関連付ける。レコーダ206が情報を少なくとも1つのコピーされたパケット、例えばタイムスタンプまたはデータ長インジケータと関連付ける実施形態に対して、レコーダ206は、この情報を、パケットに加えて記録されたプロトコルデータストリーム210に埋め込み得、またはレコーダ206は、この情報を直接にパケットに埋め込み得、またはレコーダ206は、パケットおよび記録されたプロトコルデータストリーム210から分離したロケーションにアソシエーションを格納し得る。

30

【0038】

40

図2に図示されるように、レコーダ206は、第1のデバイス202上に存在し得る。図3は、レコーダ206が第2のデバイス上に存在する実施形態を図示し、図3中においてレコーダ206は、第2のデバイス212上に存在する。図4は、レコーダ206が第3のデバイス上に存在する実施形態を図示する。レコーダ206が存在し得るデバイスは、クライアントコンピューティングシステムと、サーバコンピューティングシステムと、プロキシサーバコンピューティングシステムと、ネットワークパケットスニフリングコンピューティングシステムと、プロトコルアナライザコンピューティングシステムと、パススルサーバコンピューティングシステムとを含む。

【0039】

レコーダ206は、少なくとも1つのコピーされたパケットと、一部の実施形態におい

50

ては少なくとも1つのコピーされたパケットに関連付けられた情報とを用いて、記録されたプロトコルデータストリーム210を作成する。一部の実施形態において、レコーダ206は、プロトコルデータストリーム210の記録を、作成した後に格納する。これらの実施形態の一部において、レコーダ206は、格納要素218にプロトコルデータストリーム210の記録を格納する。格納要素218は、固定ストレージを含み、例えばハードドライブ、フロッピー（登録商標）ドライブ、CD-RW、DVD-RWまたは、任意の他のデバイスであり、それらは電力が消えたときにデータの状態を維持する。他の実施形態において、格納要素は、1つ以上の揮発性メモリ要素を含み得、例えば、スタティックランダムアクセスメモリ（SRAM）、バースト（Burst）SRAMまたはSync Burst SRAM（BSRAM）、ダイナミックランダムアクセスメモリ（DRAM）、ファストページモードDRAM（FPM DRAM）、エンハンスドDRAM（EDRAM）、エクステンデッドデータ出力RAM（EDO RAM）、エクステンデッドデータ出力DRAM（EDO DRAM）、バーストエクステンデッドデータ出力DRAM（BEDO DRAM）、エンハンスドDRAM（EDRAM）、シンクロノスDRAM（SDRAM）、JEDEC SRAM、PC100 SDRAM、ダブルデータレートSDRAM（DDR SDRAM）、エンハンスドSDRAM（ESDRAM）、Sync Linc DRAM（SLDRAM）、ディレクトランバスDRAM（DRDRAM）、またはFerroelectric RAM（FRAM）などである。

10

【0040】

一実施形態において、格納要素は、ネットワーク格納デバイスを含む。格納要素218は、第1のデバイス202上、または第2のデバイス212上に存在し得る。他の実施形態において、格納要素218は、第3のデバイス上に存在し得、例えば、プロキシサーバコンピューティングデバイスまたはパススルーサーバコンピューティングデバイスである。さらに他の実施形態において、格納要素218は、ネットワーク上に存在し、レコーダ206は、ネットワークを介して格納要素218にアクセスし、プロトコルデータストリーム220の記録を格納する。他の実施形態において、レコーダ206は、レコーダ206が存在する同一のデバイス上にプロトコルデータストリームの記録を格納する。

20

【0041】

ここで図5を参照して、ブロック図は、プロトコルエンジン502を図示し、プロトコルエンジン502は、プロトコルデータストリーム506の記録から少なくとも1つのコピーされたパケットを読み取り、少なくとも1つのコピーされたパケットに関連付けられた情報を用いてプロトコルデータストリーム506によって表されるディスプレイデータを再生成する。プロトコルエンジン502は、プロトコルデータストリーム506を受信する。一部の実施形態において、プロトコルエンジン502は、格納要素504からプロトコルデータストリーム506を検索する。他の実施形態において、プロトコルエンジン502は、レコーダ206からプロトコルデータストリーム506を検索する。さらに他の実施形態において、プロトコルエンジン502は、別のコンピューティングデバイスからプロトコルデータストリーム506を検索する。

30

【0042】

一部の実施形態において、プロトコルエンジン502は、パケットリーダ508およびディスプレイデータ再生成要素510を含む。これらの実施形態において、パケットリーダ508は、プロトコルデータストリーム506の記録から少なくとも1つのコピーされたパケットを読み取る。一部の実施形態において、パケットリーダ508は、プロトコルデータストリーム506の記録から続けて少なくとも1つのコピーされたパケットを読み取る。

40

【0043】

プロトコルエンジン502は、少なくとも1つのコピーされたパケットおよび少なくとも1つのコピーされたパケットに関連付けられたあらゆる情報を処理する。プロトコルエンジン502は、一部の実施形態において、処理のためにディスプレイデータ再生成要素510を用いる。パケットは、ユーザに提示される認知が可能なディスプレイの再生成を

50

可能にするデータを含む。一部の実施形態において図 2 に示されるように、第 2 のデバイス 2 1 2 は、このデータを処理する。一実施形態において、処理することは、少なくとも 1 つのコピーされたパケットのコンテンツをバッファにレンダリングする (r e n d e r) ことを含む。別の実施形態において、処理することは、少なくとも 1 つのコピーされたパケットのコンテンツを認知が可能な仕方ではレンダリングすることを含む。再生成されたディスプレイは、聴覚的、視覚的、触覚的、または嗅覚的な提示、またはこれらの組み合わせを含み、それらに限定はされない。

【 0 0 4 4 】

一部の実施形態において、プロトコルエンジン 5 0 2 は、第 1 のデバイス 2 0 2 上に存在する。他の実施形態において、プロトコルエンジン 5 0 2 は、第 2 のデバイス 2 1 2 上に存在する。さらに他の実施形態において、プロトコルエンジンは、第 3 のデバイス上に存在し、例えば、プロキシサーバコンピューティングデバイスまたはバススルーサーバコンピューティングデバイスに存在する。

【 0 0 4 5 】

ここで図 7 に進んで参照すると、ブロック図は、元来は図 2 におけるレコーダ 2 0 6 として記述されたレコーダ 7 0 2 のさらなる詳細を図示する。簡潔に概略すると、レコーダ 7 0 2 は、プロトコルデータストリーム 7 1 0 のインターセプションとプロトコルデータストリーム 7 1 0 の記録 7 1 2 の作成とを介してサーバから生成されるデータを記録する。

【 0 0 4 6 】

レコーダ 7 0 2 は、一実施形態においてプロトコルデータストリームインターセプタ 7 0 4 と、パケットコピア 7 0 6 と、記録ジェネレータ 7 0 8 とを含む。一実施形態において、レコーダ 7 0 2 は、プロトコルデータストリームインターセプタ 7 0 4 を用いてプロトコルデータストリーム 7 1 0 をモニターする。別の実施形態において、レコーダ 7 0 2 は、プロトコルデータストリームインターセプタ 7 0 2 を用いて、第 1 のデバイス 2 0 2 から第 2 のデバイス 2 1 2 に送信される複数のパケットを含むプロトコルデータストリーム 7 1 0 をインターセプトする。パケットコピア 7 0 6 は、プロトコルデータストリームの少なくとも 1 つのパケットをコピーする。パケットコピア 7 0 6 は、プロトコルデータストリームの中のパケットをコピーするか否かを決定する。一部の実施形態において、パケットコピア 7 0 6 は、ポリシーに回答してこの決定を下す。これらの実施形態において、パケットコピア 7 0 6 は、パケットがなんらかのデータを含むか否か、またはパケットの中に含まれるデータの種類の基に基づいて、パケットをコピーすることを決定し得る。

【 0 0 4 7 】

一実施形態において、レコーダ 7 0 2 は、記録ジェネレータ 7 0 8 を利用して、少なくとも 1 つのコピーされたパケットを用いてプロトコルデータストリームの記録を作成する。記録ジェネレータは、少なくとも 1 つのコピーされたパケットをプロトコルデータストリーム 7 1 0 の記録 7 1 2 の中にアセンブルする。一部の実施形態において、記録ジェネレータ 7 0 8 は、プロトコルデータストリームの記録の中に情報を埋め込む。この情報は、パケット内に含まれるデータによって表されるディスプレイデータをいつ再生成するかを示す時間基準、パケット内に含まれるデータを記述したデータ長インジケータ、またはプロトコルデータストリーム 7 1 0 内に含まれるデータによって表されるディスプレイデータを再生成するために用いられる他の種類の情報を含み得、それらに限定はされない。

【 0 0 4 8 】

図 8 は、プロトコルデータストリーム 7 1 0 の記録 7 1 2 の一実施形態を図示する。示される実施形態において、記録ジェネレータ 7 0 8 は、遠隔提示プロトコルデータチャンクの中に少なくとも 1 つのコピーされたパケットを分類した。記録ジェネレータ 7 0 8 は、時間基準およびデータ長を各遠隔提示プロトコルデータチャンクと関連付け、プロトコルデータストリーム 7 1 0 の記録 7 1 2 の中に情報およびパケットをアセンブルした。

【 0 0 4 9 】

図 7 に戻って参照すると、一実施形態において、レコーダ 7 0 2 は、格納要素 7 1 4 に

10

20

30

40

50

プロトコルデータストリーム 710 の完了した記録 712 を格納する。一部の実施形態において、格納要素は、ネットワーク上に位置し、レコーダ 702 は、ネットワークを介して格納要素 714 に記録 712 を送信する。他の実施形態において、格納要素は、プロキシサーバコンピューティングデバイス上に位置する。さらに他の実施形態において、格納要素は、パススルーサーバコンピューティングデバイス上に位置する。一部の実施形態において、格納要素 714 は、レコーダ 702 と同一のデバイス上に存在する。

【0050】

一実施形態において、図 7 によって陰影で図示されるプロトコルデータストリームの記録およびプレイバックのためのシステムは、レコーダ 702 および図 5 で議論されるプレイバックデバイス 514 を含む。プレイバックデバイス 514 は、プロトコルエンジン 502 を含み、該プロトコルエンジン 502 は、パケットリーダ 508 を用いてパケットデータストリームの記録 712 から少なくとも 1 つのコピーされたパケットを受信し、読み取り、埋め込まれた情報を用いてプロトコルデータストリームの記録 712 によって表されるディスプレイデータを再生成する。一部の実施形態において、プロトコルエンジン 502 は、ディスプレイデータの再生成中に続けてパケットを読み取る。

【0051】

図 7 によって図示される別の実施形態において、サーバから生成されたデータを記録および再生するためのシステムは、レコーダ 702 と、格納要素 714 と、プロトコルエンジン 502 とを含む。レコーダ 702 は、プロトコルデータストリームの記録を生成し、記録 712 を格納要素 714 に格納する。レコーダはプロトコルデータストリームから少なくとも 1 つのパケットをコピーし、時間基準またはパケット内のデータの長さを記述するデータ長インジケータを含むが、それらには限定されない少なくとも 1 つのパケットと情報とを関連付ける。プロトコルエンジン 502 は、格納要素 714 と通信し、プロトコルデータストリームの記録から少なくとも 1 つのパケットを読み取り、少なくとも 1 つのパケットと関連付けられた情報を用いてプロトコルデータストリーム 710 の記録によって表されるディスプレイデータを再生成する。

【0052】

一実施形態において、レコーダ 702、プロトコルエンジン 502、または格納要素 714 は、第 1 のデバイス 202 上に共に、または別個に位置し得る。他の実施形態において、それらは、第 2 のデバイス 212 上に共に、または別個に位置し得る。さらに別の実施形態において、それらは第 3 のデバイス、例えばプロキシサーバコンピューティングデバイス、ネットワークパケットスニフファ、またはパススルーサーバコンピューティングデバイスの上に共に、または別個に存在し得る。さらなる他の実施形態において、格納要素 714 は、レコーダ 702 およびプロトコルエンジン 502 とは分離して、格納領域ネットワーク上に存在し得る。

【0053】

ここで図 6 を参照して、流れ図は、サーバから生成されるデータを記録および再生するための方法を要約する。簡潔に概略すると、レコーダ 206 は、第 1 のデバイス 202 から第 2 のデバイス 212 に送信される複数のパケットを含むプロトコルデータストリーム 208 をインターセプトする（ステップ 602）。レコーダ 206 は、プロトコルデータストリーム 208 から少なくとも 1 つのパケットをコピー（ステップ 604）し、プロトコルデータストリーム 210 の記録を作成する（ステップ 606）。該記録は、プロトコルエンジン 502 が後程に、記録されたプロトコルデータストリーム 210 によって表されるディスプレイデータを再生成するときに用いる（ステップ 608、ステップ 610）。

【0054】

レコーダ 206 は、複数のパケットを含むプロトコルデータストリーム 208 をインターセプトし、第 1 のデバイス 202 から第 2 のデバイス 212 に送信されるディスプレイデータを表す。レコーダ 206 は、プロトコルデータストリーム 208 の少なくとも 1 つのパケットをコピーする。レコーダ 206 は、少なくとも 1 つのコピーされたパケットを

用いてプロトコルデータストリームの記録を作成する。レコーダ 206 は、一部の実施形態において情報を少なくとも 1 つのコピーされたパケットと関連付ける。情報は、タイムスタンプまたはデータ長インジケータを含み得る。これらの実施形態の一部において、レコーダ 206 は、パケットと関連付けられた情報をプロトコルデータストリーム 210 の記録に埋め込む。他のこれらの実施形態において、レコーダ 206 は、パケットと関連付けられた情報を別のプロトコルデータストリームに格納する。さらに他のこれらの実施形態において、レコーダは、パケットと関連付けられた情報をデータストアに格納する。プロトコルエンジン 502 は、プロトコルデータストリーム 210 の記録から少なくとも 1 つのコピーされたパケットを読み取り、少なくとも 1 つのコピーされたパケットと関連付けられた情報を用いてプロトコルデータストリーム 210 によって表されるディスプレイデータを再生成する。

10

【0055】

ここで図 11 に進んで参照すると、ブロック図は、ステートフル遠隔提示プロトコルのプレイバックの間にリアルタイムをシーキングするためのシステムを図示する。簡潔に概略すると、この図面は、プレイバックデバイス 514 (上記の図 5 を参照) の実施形態を図示し、該プレイバックデバイス 514 は、2 つのプロトコルエンジン 502、バックグラウンドプロトコルエンジン 1102 およびフォアグラウンドプロトコルエンジン 1106、ならびにステートスナップショット 1104 およびディスプレイ 1108 を含む。バックグラウンドプロトコルエンジン 1102 は、プロトコルデータストリーム 1110 の記録を受信し、プロトコルデータストリーム 1110 の記録を読み取る。該記録は、複数のパケットを含み、ディスプレイデータを表す。一実施形態において、プレイバックデバイス 514 は、少なくとも 1 つのパケットのコンテンツをプロトコルデータストリーム 1110 にレンダリングすることと、ディスプレイ 1108 を用いて結果を表示することとによってディスプレイデータを再生成する。結果は、認知が可能な聴覚的、視覚的、触覚的、または嗅覚的な提示を含み、それらに限定はされない。

20

【0056】

ここで図 11 を参照して、より詳細な点に関して、バックグラウンドプロトコルエンジン 1102 は、レンダリングされたディスプレイデータの受信者が、プロトコルデータストリーム 1110 の提示の間にリアルタイムにおけるコンテンツをシーキングすることを可能にする。バックグラウンドプロトコルエンジン 1102 は、少なくとも 1 つのステートスナップショット 1104 を生成し、その一方でプロトコルデータストリーム 1110 から少なくとも 1 つのパケットを読み取る。一実施形態において、バックグラウンドプロトコルエンジン 1102 は、バッファに少なくとも 1 つのパケットのコンテンツをレンダリングする。この実施形態において、バッファは、オフスクリーンバッファを含み得る。この実施形態において、バックグラウンドプロトコルエンジン 1102 は、少なくとも 1 つのパケットのコンテンツをレンダリングしながら、少なくとも 1 つのステートスナップショット 1104 を生成する。バックグラウンドプロトコルエンジン 1102 は、フォアグラウンドプロトコルエンジン 1106 に対してステートスナップショット 1104 を利用可能にする。

30

【0057】

ステートスナップショット 1104 は、ディスプレイデータの再生成を可能にするが、なぜならば、レコーダ 206 がプロトコルデータストリーム 208 からプロトコルデータストリーム 1110 の記録に少なくとも 1 つのパケットをコピーした時点で、プロトコルデータストリーム 1110 をレンダリングするプロトコルエンジンの状態を格納するからである。一実施形態において、ステートスナップショット 1104 は、ある時点のスクリーンの状態を記述するデータ構造を含む。別の実施形態において、ステートスナップショット 1104 は、プロトコルデータストリーム 1110 における基準点でプロトコルエンジンの状態を構成する全ての変数、イメージおよびデータ構成要素を表す。フォアグラウンドプロトコルエンジン 1106 もまた、プロトコルデータストリーム 1110 の記録を受信し、プロトコルデータストリーム 1110 を元来レンダリングしたプロトコルエンジ

40

50

ンの状態を再び作成することによって、プロトコルデータストリーム 1 1 1 0 の中に少なくとも 1 つのパケットのコンテンツをレンダリングする。一実施形態において、フォアグラウンドプロトコルエンジン 1 1 0 6 は、ステートスナップショット 1 1 0 4 のコンテンツを用いて少なくとも 1 つのパケットのコンテンツをレンダリングする。

【 0 0 5 8 】

一実施形態において、ステートスナップショット 1 1 0 4 は、データ構造を含む。他の実施形態において、ステートスナップショット 1 1 0 4 は、データベースを含む。一実施形態において、ステートスナップショット 1 1 0 4 のコンテンツは、可視の表面の状態に関するディスプレイデータを含む。別の実施形態において、ステートスナップショット 1 1 0 4 のコンテンツは、オフスクリーンの表面の状態に関するディスプレイデータを含む。さらなる別の実施形態において、ステートスナップショット 1 1 0 4 のコンテンツは、描画対象の状態に関するディスプレイデータを含む。一部の実施形態において、ステートスナップショット 1 1 0 4 のコンテンツは、カラーパレットの状態に関するディスプレイデータを含む。他の実施形態において、ステートスナップショット 1 1 0 4 のコンテンツは、キャッシュされた対象物の状態に関するディスプレイデータを含む。さらに他の実施形態において、ステートスナップショット 1 1 0 4 のコンテンツは、バッファの状態に関するディスプレイデータを含む。

【 0 0 5 9 】

フォアグラウンドプロトコルエンジン 1 1 0 6 は、プロトコルデータストリーム 1 1 1 0 の記録を受信し、ステートスナップショット 1 1 0 4 を用いて要求されたデジタルデータの表現 (r e p r e s e n t a t i o n) を含むパケットを識別し、パケットをレンダリングする。一部の実施形態において、フォアグラウンドプロトコルエンジン 1 1 0 6 は、ディスプレイ 1 1 0 8 を用いるビューアへの提示のために、プロトコルデータストリーム 1 1 1 0 の記録のリアルタイムにおける認識できる表現を生成する。一部の実施形態において、フォアグラウンドプロトコルエンジン 1 1 0 6 は、プロトコルデータストリーム 1 1 1 0 に少なくとも 1 つのパケットのコンテンツをレンダリングすることによって、リアルタイムにおける認識できる表現を生成する。認識できる表現は、聴覚的、視覚的、触覚的、または嗅覚的な提示を、別個にまたは共に含み得、それらに限定はされない。

【 0 0 6 0 】

フォアグラウンドプロトコルエンジン 1 1 0 6 がプロトコルデータストリーム 1 1 1 0 に少なくとも 1 つのパケットのコンテンツをレンダリングする実施形態の 1 つにおいて、フォアグラウンドプロトコルエンジン 1 1 0 6 は、バックグラウンドプロトコルエンジン 1 1 0 2 によってレンダリングすることと同時に、プロトコルデータストリーム 1 1 1 0 に少なくとも 1 つのパケットのコンテンツをレンダリングすることを開始する。しかし、バックグラウンドプロトコルエンジン 1 1 0 2 は、バッファのみにレンダリングし、フォアグラウンドプロトコルエンジン 1 1 0 6 によって開始されたリアルタイムにおける認知可能なレンダリングの完了に先立って、レンダリングおよび少なくとも 1 つのステートスナップショット 1 1 0 4 の生成を完了し、一実施形態において、フォアグラウンドプロトコルエンジン 1 1 0 6 は、バッファおよび認知可能な仕方の両方にレンダリングする。一実施形態において、バックグラウンドプロトコルエンジン 1 1 0 2 は、最大限に可能な限りの速度でプロトコルデータストリーム 1 1 1 0 をレンダリングし、さもなければレンダリングのための時間を特定し得る記録に関連付けられたあらゆるタイムスタンプに関係なくなされる。従って、プロトコルデータストリーム 1 1 1 0 の記録のリアルタイムにおける認知可能な表現の生成の間に、少なくとも 1 つのステートスナップショット 1 1 0 4 は、フォアグラウンドプロトコルエンジン 1 1 0 6 に利用可能である。

【 0 0 6 1 】

一実施形態において、フォアグラウンドプロトコルエンジン 1 1 0 6 は、連続した仕方でプロトコルデータストリーム 1 1 1 0 の記録内で複数のパケットのコンテンツをレンダリングする。この実施形態において、ユーザにレンダリングおよび提示されたディスプレイデータは、プロトコルデータストリームが記録された時間が発生した順番でディスプレ

イを提示する。プロトコルデータストリーム 1 1 1 0 の記録は、例えばタイムスタンプなどの情報を含み、フォアグラウンドプロトコルエンジン 1 1 0 6 がディスプレイデータを連続してレンダリングするときに用いられる。一部の実施形態において、フォアグラウンドプロトコルエンジン 1 1 0 6 は、リアルタイムにおいてディスプレイデータをレンダリングする。フォアグラウンドプロトコルエンジン 1 1 0 6 が、プロトコルデータストリーム 1 1 1 0 の記録における特定の packets によって表される特定のディスプレイデータを再生成する要求を受信するときに、フォアグラウンドプロトコルエンジン 1 1 0 6 は、識別されたステートスナップショット 1 1 0 4 のコンテンツを用いて要求されたディスプレイデータをレンダリングする。

【 0 0 6 2 】

一部の実施形態において、バックグラウンドプロトコルエンジン 1 1 0 2 およびフォアグラウンドプロトコルエンジン 1 1 0 6 は、同一のデバイス上に存在する。他の実施形態において、バックグラウンドプロトコルエンジン 1 1 0 2 およびフォアグラウンドプロトコルエンジン 1 1 0 6 は、別個のデバイス上に存在する。

【 0 0 6 3 】

ここで図 9 に戻って参照すると、流れ図は、ステートフル遠隔提示プロトコルのプレイバックの間のリアルタイムシーキングのための方法で取られるステップの一実施形態を図示する。簡潔に概略すると、プロトコルデータストリームの記録における packets のコンテンツによって表されるデータディスプレイのレンダリングに対する要求がある（ステップ 9 0 2）。まず要求された packets に関連付けられたタイムスタンプよりも後ではない関連付けられたタイムスタンプを有するステートスナップショットを識別（ステップ 9 0 4）し、識別されたステートスナップショットに応答する要求されたコンテンツをレンダリングする（ステップ 9 0 6）ことによって、適切な packets のコンテンツはレンダリングされる。

【 0 0 6 4 】

一実施形態において、フォアグラウンドプロトコルエンジン 1 1 0 6 は、プロトコルデータストリーム 1 1 1 0 のレコーディングにおける packets のコンテンツをレンダリングする要求を受信する。プロトコルデータストリーム 1 1 1 0 は、複数の packets を含み、該 packets のコンテンツは、ディスプレイデータを表す。一部の実施形態において、フォアグラウンドプロトコルエンジン 1 1 0 6 が、ディスプレイ 1 1 0 8 を用いるビューアにプロトコルデータストリーム 1 1 1 0 の記録における packets のコンテンツをレンダリングすることによってディスプレイデータを再生成し、ビューアが特定のディスプレイデータをシーキングすることを望むときに、要求は結果として起こる。

【 0 0 6 5 】

フォアグラウンドプロトコルエンジン 1 1 0 6 は、要求された packets に関連付けられたタイムスタンプよりも後ではない関連付けられたタイムスタンプを有するステートスナップショット 1 1 0 4 を識別する。フォアグラウンドプロトコルエンジン 1 1 0 6 は、識別されたステートスナップショット 1 1 0 4 に応答する、要求された packets のコンテンツによって表されるディスプレイデータを表示する。一実施形態において、識別されたステートスナップショット 1 1 0 4 は、プロトコルデータストリーム 1 1 1 0 から正確な packets を示し、プロトコルデータストリーム 1 1 1 0 のコンテンツは、フォアグラウンドプロトコルエンジン 1 1 0 6 が、要求されたディスプレイデータをユーザに提供するためにレンダリングし得る。

【 0 0 6 6 】

他の実施形態において、識別されたステートスナップショット 1 1 0 4 は、レコーダがプロトコルデータストリーム 1 1 1 0 から packets をコピーしたが、コピーされた packets のコンテンツによって表されるディスプレイデータが、ビューアによって要求されたディスプレイデータに先んずる時点における、プロトコルデータストリームをレンダリングする、プロトコルエンジンの状態を含む。これらの実施形態の一部において、ステートスナップショットと要求されたディスプレイデータの表現を含む packets との間に複数のパ

10

20

30

40

50

ケットがある。これらの実施形態の一部において、フォアグラウンドプロトコルエンジン 1106 は、オフスクリーンバッファのみに中間のケットまたは複数の中間のケットのコンテンツをレンダリングする。フォアグラウンドプロトコルエンジン 1106 は、次いで、コンテンツが、認知可能な仕方ではオフスクリーンバッファとユーザとの両方にディスプレイデータを表すケットをレンダリングする。一実施形態において、フォアグラウンドプロトコルエンジン 1106 は、要求されたコンテンツによって表されるディスプレイデータに先立って、認知可能な仕方では中間のケットのコンテンツによって表されるディスプレイデータを提示する。

【0067】

ここで図 10 を参照して、流れ図は、遠隔提示プロトコルのプレイバックの間にリアルタイムにおけるシーキングを可能にするステートスナップショットを生成するために取られるステップの一実施形態を図示する。簡潔に概略すると、バックグラウンドプロトコルエンジン 1102 は、プロトコルデータストリーム 1110 の記録を受信し、プロトコルデータストリーム 1110 内の複数のケットのコンテンツによって表されるディスプレイデータを再生成する間に少なくとも 1 つのステートスナップショットを生成する。

【0068】

バックグラウンドプロトコルエンジン 1102 は、複数のケットを含むプロトコルデータストリーム 1110 の記録を受信（ステップ 1002）する。バックグラウンドプロトコルエンジン 1102 は、プロトコルデータストリームの記録の表現を生成する。一実施形態において、バックグラウンドプロトコルエンジン 1102 は、複数のケットのコンテンツをバッファにレンダリングすることによって表現を生成する。一部の実施形態において、バッファは、オフスクリーンバッファである。

【0069】

一部の実施形態において、フォアグラウンドプロトコルエンジン 1106 もまた、プロトコルデータストリーム 1110 の記録を受信する。これらの実施形態において、フォアグラウンドプロトコルエンジン 1106 は、プロトコルデータストリームの記録の人間に認知可能な表現を生成するが、上述のように、フォアグラウンドプロトコルエンジン 1106 は、オフスクリーンバッファと認知可能な仕方との両方にレンダリング（ステップ 1004）する。一つのこれらの実施形態において、フォアグラウンドプロトコルエンジン 1106 は、複数のケットのコンテンツをレンダリングすることによってプロトコルデータストリーム 1110 の記録の人間に認知可能な表現を生成し、実質的に同時に、バックグラウンドプロトコルエンジン 1102 が、プロトコルデータストリームの記録の間に少なくとも 1 つのステートスナップショットを生成する。

【0070】

プロトコルデータストリーム 1110 の記録における少なくとも 1 つのケットの読み取りの後に、バックグラウンドプロトコルエンジン 1102 は、少なくとも 1 つのステートスナップショットを生成（ステップ 1006）する。一実施形態において、バックグラウンドプロトコルエンジン 1102 は、プロトコルデータストリーム 1110 の記録の連続する読み取りの間に少なくとも 1 つのステートスナップショットを生成する。別の実施形態において、バックグラウンドプロトコルエンジン 1102 は、プロトコルデータストリーム 1110 の記録における少なくとも 1 つのケットを読み込み、実質的に同時に、バッファへのケットのコンテンツのレンダリングを行う。一実施形態において、バックグラウンドプロトコルエンジン 1102 は、次いで生成されたステートスナップショット 1104 を格納（ステップ 1008）する。バックグラウンドプロトコルエンジン 1102 が定期的に複数のステートスナップショットを生成する実施形態において、ステートスナップショットは、プロトコルデータストリーム 1110 の記録の間じゅう、マーカーとして機能し得、プロトコルデータストリーム 1110 における特定の時点のロケーションおよびステートスナップショット 1104 の前または後に到着するケットのロケーションを補助する。

【0071】

図 1 2 を参照すると、流れ図は、ステートスナップショットの適応的な生成のためにとられるステップの一実施形態を描いている。簡潔に概略すると、バックグラウンドプロトコルエンジン 1 1 0 2 は、ビューアの活動をモニタリングし、ビューアの活動のレベルに応じて一つ以上のステートスナップショット 1 1 0 4 を生成する。

【 0 0 7 2 】

ユーザに対してプロトコルデータストリーム 1 1 1 0 の記録の表示を提示する間（ステップ 1 2 0 2 ）、バックグラウンドプロトコルエンジン 1 1 0 2 は、ユーザの活動をモニタリングする（ステップ 1 2 0 4 ）。一実施形態において、フォアグラウンドプロトコルエンジン 1 1 0 6 は、プロトコルデータストリーム 1 1 1 0 の記録の表示を生成し、ディスプレイ 1 1 0 8 を用いて、その表示をユーザに提示する。他の実施形態において、バックグラウンドプロトコルエンジン 1 1 0 2 は表示を生成する。さらに他の実施形態において、第 3 のデバイスが表示を生成する。

【 0 0 7 3 】

バックグラウンドプロトコルエンジン 1 1 0 2 は、提示の間、ユーザの活動をモニタリングする（ステップ 1 2 0 4 ）。ユーザの活動をモニタリングすることによって、バックグラウンドプロトコルエンジン 1 1 0 2 は、活動のモニタリングに応答して活動のプロフィールを作成する（ステップ 1 2 0 6 ）。バックグラウンドプロトコルエンジンは、作成された活動プロフィールに応答して、少なくとも一つのステートスナップショット 1 1 0 4 を生成する（ステップ 1 2 0 8 ）。

【 0 0 7 4 】

一部の実施形態において、バックグラウンドプロトコルエンジン 1 1 0 2 は、ユーザの活動レベルを識別する。一部の実施形態において、バックグラウンドプロトコルエンジン 1 1 0 2 は、活動のない期間を識別する。他の実施形態において、バックグラウンドプロトコルエンジン 1 1 0 2 は、表示データ内のユーザにとって関心のあるエリアを識別する。活動プロフィールは、これらの識別を反映する。

【 0 0 7 5 】

バックグラウンドプロトコルエンジン 1 1 0 2 は、活動プロフィールに応答して、少なくとも一つのステートスナップショットを生成する。一部の実施形態において、バックグラウンドプロトコルエンジン 1 1 0 2 は、一つ以上のステートスナップショットの間隔を拡張することを決定する。他の実施形態において、バックグラウンドプロトコルエンジン 1 1 0 2 は、一つ以上のステートスナップショットの間隔を減少させることを決定する。さらに他の実施形態において、バックグラウンドプロトコルエンジン 1 1 0 2 は、活動プロフィールに応答して、少なくとも一つのステートスナップショットを除去することを決定する。さらに他の実施形態において、バックグラウンドエンジンプロトコル 1 1 0 2 は、活動プロフィールに応答して、少なくとも一つのステートスナップショットを追加することを決定する。

【 0 0 7 6 】

一実施形態において、バックグラウンドプロトコルエンジン 1 1 0 2 は、シーキング確率の見込みの統計的な分布を識別する。図 1 3 は、一実施形態のシーキング確率の見込みの統計的な分布の三つのタイプを描いている図である。この実施形態において、バックグラウンドプロトコルエンジン 1 1 0 2 は、ユーザによって行なわれたシーキング要求についてのデータを収集し、格納する。一実施形態において、データは、どの程度規則的に、ユーザがシーキング要求を行なうかを含む。一実施形態において、データは、各シーキング要求の範囲、要求された表示データと、プロトコルデータストリーム 1 1 1 0 の記録内のパケットのコンテンツをレンダリングすることによって、ユーザに提示される現在の表示データとの間の距離を含む。範囲は、時間単位か、またはプロトコルデータストリーム 1 1 1 0 の全記録の長さに関してかで、記述され得る。一実施形態において、シーキング要求が行なわれるタイムスタンプが記録される。

【 0 0 7 7 】

図 1 4 は、シーキング確率分布の決定に応答して、ステートスナップショットを生成す

る一実施形態を描いている図である。バックグラウンドプロトコルエンジン 1102 は、収集されたシーキング要求データを使用して、現在提示されている表示データを中央に置く、シーキング確率分布のグラフを生成する。バックグラウンドプロトコルエンジン 1102 は、ストリーム内の各位置に、ユーザがその位置に関連する表示データをシーキングすることを要求する確率の見込みを指している値を割り当てる。このデータを使用して、バックグラウンドプロトコルエンジン 1102 は、生成されたスナップショット 1104 をどこに配置するべきかを決定し、シーキング確率の統計的な分布に応答して、少なくとも一つのステートスナップショット 1104 を生成する。

【0078】

図 15 は、ユーザの使用パターンの一実施形態を描いている。一実施形態において、バックグラウンドプロトコルエンジン 1102 は、ユーザの使用パターンに基づいて、ユーザに関する活動プロフィールを形成する。使用パターンは、識別されたシーキング確率を反映する。シーキング確率の高いエリアは、ステートスナップショットの密度が高く、シーキング確率の低いエリアは、ステートスナップショットの密度が低い。一部の実施形態において、ステートスナップショットの任意の対の間の距離は、それらの間の平均的なシーキング確率に対して反比例する。バックグラウンドプロトコルエンジン 1102 は、ユーザが確率の高いエリアをシーキングすることを予期し、それにより、シーキングの大部分が速くなる。なぜならば、生成されたステートスナップショット 1104 の間隔が比較的短いからである。個々のシーキング要求が、あまり遅くならないことを確実にするために、一実施形態において、シーキング確率が非常に低いときでさえ、バックグラウンドプロトコルエンジン 1102 は、生成されたステートスナップショット 1104 の間隔に上限を設ける。同様に、別の実施形態において、下限は、非常に確率の高いエリアに、かなり密接してステートスナップショットを配置することを防止する。一部の実施形態において、待ち時間を最小化するために、ステートスナップショットの配置を決定するときに、隣接するステートスナップショット間のレンダリングの量が考慮される。

【0079】

新たなユーザまたは使用パターンを区別することのできないユーザを有する実施形態に関して、バックグラウンドプロトコルエンジン 1102 は、デフォルトのステートスナップショット生成パターンを適用する。このパターンは、シーキングすることのほとんどが、現在のフレームの付近で各方向に発生することを想定するが、長い範囲のシーキングの性能は、せいぜい満足できる程度であり得る。多くの小さなシーキングステップとして、現在のフレームの周囲を前後にジョギングすることが、ジョグホイール入力デバイスを用いて達成され得る時には、一般的なユーザは、高い性能を要求する。長い範囲をシーキングすることは、あまり一般的ではなく、顕著な遅れは、許容可能なトレードオフであり得る。

【0080】

ユーザが識別された使用パターンからはずれる場合には、バックグラウンドプロトコルエンジン 1102 は、ユーザの知識を用いることなくライブプレイバックの間、ステートスナップショット生成パターンを調整する。バックグラウンドプロトコル 1102 は、ステートスナップショットの位置を移動させ、新しい使用パターンのために調整する。例えば、マウスホイールを用いた小さなステップにおいて、普通にシーキングするユーザが、長い範囲をシーキングすることを開始する場合には、バックグラウンドプロトコルエンジン 1102 は、現在のフレームの周囲のステートスナップショットの数を低減させ、長い範囲におけるエリア内のステートスナップショットを追加するための供給源を使用できるようにする。

【0081】

図 16 は、ステートスナップショットの適応性の生成のためのシステムにおいて使用される、上記の方法の一実施形態を要約し、該ステートスナップショットは、バックグラウンドプロトコルエンジン 1602 と、フォアグラウンドプロトコルエンジン 1608 と、プロトコルデータストリーム 1612 と、活動プロフィール 1604 と、ステートスナッ

ブショット 1 6 0 6 とを含む。フォアグラウンドプロトコルエンジン 1 6 0 8 は、プロトコルデータストリームの記録の表示をビューアに対して提示する。バックグラウンドプロトコルエンジン 1 6 0 2 は、提示の間、ビューアの活動をモニタリングし、モニタリングに回答して、活動プロフィール 1 6 0 4 を形成し、活動プロフィールに回答して、ステータスナッシュショット 1 6 0 6 を生成し、かつ維持する。

【 0 0 8 2 】

図 1 8 を参照すると、流れ図は、記録されたコンピュータセッションのプレイバックのためのプレイバック命令を生成するためにとられるステップの一実施形態を描いている。簡潔に概略すると、第 1 のデバイス上で実行しているプロトコルエンジンは、記録されたセッションを受信する（ステップ 1 8 0 2）。記録されたストリームは、第 2 のデバイス上で実行されるアプリケーションプログラムによって生成された表示データを表している複数のパケットを含む。プロトコルエンジンは、人間が認識できる方法（human-perceptible manner）でパケットのコンテンツをレンダリングするように、記録されたストリーム内のパケットのために決定を行う（ステップ 1 8 0 4）。次に、プロトコルエンジンは、プレイバックデータ構造内に決定を格納する（ステップ 1 8 0 6）。

【 0 0 8 3 】

一実施形態において、プロトコルエンジンは、図 5 に記述されたように、プロトコルエンジン 5 0 2 を含む。他の実施形態において、プロトコルエンジンは、図 1 1 に記述されたようにバックグラウンドプロトコルエンジン 1 1 0 2 を含む。さらに他の実施形態において、プロトコルエンジンは、図 1 1 に記述されたように、フォアグラウンドプロトコルエンジン 1 1 0 6 を含む。プロトコルエンジンがバックグラウンドエンジン 1 1 0 2 を含む一部の実施形態において、プロトコルエンジンは、バックグラウンドプロトコルエンジン 1 1 0 2 の機能を実行することを中止し得、フォアグラウンドプロトコルエンジン 1 1 0 6 の機能を実行することを開始し得る。プロトコルエンジンがフォアグラウンドプロトコルエンジン 1 1 0 6 を含む一部の実施形態において、プロトコルエンジンは、フォアグラウンドプロトコルエンジン 1 1 0 6 の機能を実行することを中止し得、バックグラウンドプロトコルエンジン 1 1 0 2 の機能を実行することを開始する。他の実施形態において、プロトコルエンジンは、プロトコルエンジン 1 1 0 2 とフォアグラウンドプロトコルエンジン 1 0 0 6 との両方を含む。これらの実施形態のうちの一部において、バックグラウンドプロトコルエンジン 1 1 0 2 とフォアグラウンドプロトコルエンジン 1 1 0 6 とは、同じデバイスに存在する。他の実施形態において、バックグラウンドプロトコルエンジン 1 1 0 2 とフォアグラウンドプロトコルエンジン 1 1 0 6 とは、別個のデバイスに存在する。

【 0 0 8 4 】

一実施形態において、プロトコルエンジンは、人間が認識できる方法でパケットを表示するように、記録されたセッション内のパケットのために決定を行う（ステップ 1 8 0 4）。表示は、聴覚的、視覚的、触覚的、または嗅覚的な提示、あるいはそれらの組み合わせを含むが、それらには限定されない。一部の実施形態において、プロトコルエンジンは、パケットのコンテンツの応答に基づいて、パケットを表示することを決定する。これらの実施形態のうちの一部において、プロトコルエンジンは、入力フォーカスを有するアプリケーションプログラムの指標に回答して決定を行なう。これらの実施形態のうちの一部の実施形態において、プロトコルエンジンは、パケット内に格納されるユーザ入力のタイプの評価に回答して決定を行なう。これらの実施形態のうちの一部において、プロトコルエンジンは、パケットによって格納されるグラフィックの更新のタイプの評価に回答して決定を行なう。これらの実施形態のうちの一部の実施形態において、プロトコルエンジンは、パケットによって格納される相互作用シーケンスのタイプの評価に回答して決定を行なう。

【 0 0 8 5 】

一実施形態において、プロトコルエンジンは、プレイバックデータ構造内に決定を格納

する(1806)。一部の実施形態において、プレイバックデータ構造は、記録されたストリーム内に含まれる表示データをどのように再生成するのかを記述する。一実施形態において、プレイバックデータ構造内に格納される命令は、表示データをレンダリングする処理を制御する。一実施形態において、プレイバックデータ構造は、記録されたストリーム内のパケットのコンテンツをレンダリングするための時間を含む。この実施形態において、プレイバックデータ構造に含まれる時間が、パケットのコンテンツをレンダリングするために使用され、記録におけるパケットと関連するレンダリング時間は、例えあったとしても、使用されない。一実施形態において、プレイバックデータ構造は、レンダリングの時間を変更する際に、ユーザの入力を受け入れる。

【0086】

10

一部の実施形態において、プレイバックデータ構造は、記録されたセッションの一つ以上のプレイバックをどのように実行するかを記述するメタデータを含む。一実施形態において、プレイバックデータ構造は、記録ストリーム内の各パケットに関する記録からなり、相対的にどの時点において適切に、そのパケットのコンテンツが、プレイバックの間にレンダリングされるべきかを示している。一部の実施形態において、メタデータはまた、パケットの開始のファイル内にオフセットを含む。

【0087】

再び図17を参照すると、ブロック図は、記録されたセッションをレンダリングするためのシステムの一実施形態を描き、該システムは、第1のデバイス1702と、バックグラウンドプロトコルエンジン1704と、プレイバックデータ構造1706と、フォアグラウンドプロトコルエンジン1708と、ディスプレイ1710と、第2のデバイス1712と、記録されたストリーム1714とを含む。バックグラウンドプロトコルエンジン1704は、第1のデバイス1702上で実行し、記録されたストリーム1714の受信にตอบสนองして、プレイバックデータ構造1706を生成し、該記録ストリーム1714は、第2のデバイス1712または第3のデバイス上で実行されるアプリケーションプログラムによって、生成された表示データを表している。フォアグラウンドプロトコルエンジン1708は、記録されたストリーム1714を受信し、バックグラウンドプロトコルエンジン1704によって生成されるプレイバックデータ構造1706にตอบสนองして、記録されたストリーム1714をレンダリングする。

20

【0088】

30

一実施形態において、バックグラウンドプロトコルエンジン1704とフォアグラウンドプロトコルエンジン1708とのそれぞれは、記録されたストリーム1714を受信する。この実施形態において、バックグラウンドプロトコルエンジン1704は、プレイバックデータ構造を生成し、実質的に同時に、フォアグラウンドプロトコルエンジン1708は、記録されたストリームをレンダリングする。

【0089】

一実施形態において、フォアグラウンドエンジン1708は、第1のデバイス1702に存在する。図17の影の部分で示されている別の実施形態において、フォアグラウンドプロトコルエンジン1708は、第1のデバイス1702にも第2のデバイス1712にも存在しない。さらに別の実施形態において、フォアグラウンドプロトコルエンジン1708は、第3のデバイスに存在する。一部の実施形態において、フォアグラウンドプロトコルエンジン1708は、バックグラウンドプロトコルエンジン1704を含む。これらの実施形態のうちの一部において、バックグラウンドプロトコルエンジン1102とフォアグラウンドプロトコルエンジン1106とは、同じデバイスに存在する。これらの実施形態のうちの他の実施形態において、バックグラウンドプロトコルエンジン1102とフォアグラウンドプロトコルエンジン1106とは、別個のデバイスに存在する。

40

【0090】

一実施形態において、バックグラウンドプロトコルエンジンは、記録されたストリーム内の少なくとも一つのパケットをレンダリングするための少なくとも一つの命令を、プレイバックデータ構造内に格納する。別の実施形態において、バックグラウンドプロトコル

50

エンジンは、プレイバックデータ構造内にメタデータを格納する。さらに別の実施形態において、バックグラウンドプロトコルエンジンは、記録されたセッション内に少なくとも一つのパケットをレンダリングするための時間を示す記録を、プレイバックデータ構造内に格納する。

【0091】

フォアグラウンドプロトコルエンジン1708は、プレイバックデータ構造に応答して、記録されたセッション内の少なくとも一つのパケットをレンダリングする。一実施形態において、フォアグラウンドプロトコルエンジンは、記録されたセッション内の少なくとも一つのパケットを、人間が認識できる方法でバッファにレンダリングする。別の実施形態において、フォアグラウンドプロトコルエンジンは、記録されたセッション内の少なくとも一つのパケットをバッファにレンダリングする。

10

【0092】

図19を参照すると、流れ図は、記録されたコンピュータセッションをプレイバックするための方法においてとられるステップの一実施形態を描いている。簡潔に概略すると、バックグラウンドプロトコルエンジンは、第2のデバイス上で実行されるアプリケーションプログラムによって生成される表示データを表している複数のパケットを含む記録されたストリームを受信する(ステップ1902)。バックグラウンドプロトコルエンジンは、人間が認識できる方法でパケットをレンダリングするように、記録されたストリーム内の少なくとも一つのパケットのために決定を行う(ステップ1904)。バックグラウンドプロトコルエンジンは、決定をプレイバックデータ構造内に格納する(ステップ1906)。フォアグラウンドプロトコルエンジンは、記録されたストリームから少なくとも一つのパケットを引き出し(ステップ1908)、プレイバックデータ構造にアクセスし(ステップ1910)、プレイバックデータ構造に応答して、少なくとも一つのパケットをレンダリングする(ステップ1912)。

20

【0093】

一実施形態において、プロトコルエンジンは、人間が認識できる方法でパケットを表示するように、記録されたストリーム内のパケットのために決定を行う(ステップ1904)。表示は、聴覚的、視覚的、触覚的、または嗅覚的な提示、あるいはそれらの組み合わせを含むが、それらに限定されない。一部の実施形態において、プロトコルエンジンは、パケットのコンテンツに対する応答に基づいて、パケットを表示することを決定する。これらの実施形態のうちの一つにおいて、プロトコルエンジンは、入力フォーカスを有するアプリケーションプログラムの指標に応答して決定を行なう。これらの実施形態のうちの別の実施形態において、プロトコルエンジンは、パケット内に格納されるユーザの入力のタイプの評価に応答して決定を行なう。これらの実施形態のうちの一部において、プロトコルエンジンは、パケットによって格納されるグラフィックの更新のタイプの評価に応答して決定を行なう。これらの実施形態のうちの他の実施形態において、プロトコルエンジンは、パケットによって格納される相互作用シーケンスのタイプの評価に応答して決定を行なう。一実施形態において、プロトコルエンジンは、プレイバックデータ構造内に決定を格納する(1906)。

30

【0094】

一実施形態において、フォアグラウンドプロトコルエンジンは、記録されたセッションを受信する。他の実施形態において、フォアグラウンドプロトコルエンジンは、記録されたセッションを引き出す。これらの実施形態のうちの一部において、フォアグラウンドプロトコルエンジンは、記録されたセッションを格納要素から引き出す。

40

【0095】

一実施形態において、フォアグラウンドプロトコルエンジンは、少なくとも一つのパケットを記録されたストリームから引き出す(ステップ1908)。次に、この実施形態において、フォアグラウンドプロトコルエンジンは、プレイバックデータ構造にアクセスし(ステップ1910)、プレイバックデータ構造に応答して、パケットのコンテンツをレンダリングする(ステップ1912)。一部の実施形態において、プレイバックデータ構

50

造は、パケットのコンテンツを認識できる方法でレンダリングする命令を含む。これらの実施形態のうちの一つにおいて、フォアグラウンドプロトコルエンジンは、パケットのコンテンツを画面上でレンダリングする。一部の実施形態において、フォアグラウンドプロトコルエンジンは、少なくとも一つのパケットのコンテンツをバッファに必ずレンダリングする。多くの実施形態において、フォアグラウンドプロトコルエンジンが、パケットのコンテンツをバッファにレンダリングするときには、画面外バッファである。これらの実施形態のうちの一つにおいて、フォアグラウンドプロトコルエンジンは、プレイバックデータ構造によって指示されるように、パケットのコンテンツを画面外バッファにレンダリングし、パケットのコンテンツを画面上でもまたレンダリングする。

【0096】

10

他の実施形態において、プレイバックデータ構造は、パケットのコンテンツを認識できる方法でレンダリングしない命令を含む。これらの実施形態のうちの一つにおいて、プレイバックデータ構造にアクセスする際に、フォアグラウンドプロトコルは、パケットのコンテンツを認識できる方法でレンダリングしないが、パケットのコンテンツをバッファにレンダリングする。

【0097】

フォアグラウンドプロトコルエンジンが、プレイバックデータ構造に応答して、パケットのコンテンツを画面外のバッファにのみレンダリングする実施形態に関して、フォアグラウンドプロトコルエンジンは、記録されたストリームとは異なる表示データを認識できるように再生成する。一実施形態において、このことは、最初の記録されたストリームよりも短い表示データの提示をもたらす。一部の実施形態において、パケットのレンダリングされたコンテンツは、最初の表示データの再生成を簡素化する。他の実施形態において、パケットのレンダリングされたコンテンツは、表示データのカスタム化されたバージョンを提供する。一実施形態において、パケットのコンテンツを認識できる方法でレンダリングする決定は、ポリシーまたはユーザの要求に応答する。これらの実施形態は、ユーザに記録されたセッションのプレイバックに対する制御を提供する。

20

【0098】

図21を参照すると、排除される活動を含んでいない認識できる時間間隔を有する記録されたセッションのレンダリングの一実施形態が描かれている。この図において、黒いブロックは、ユーザの入力を含んでいる一つのパケットまたは複数のパケットを表し、点のあるブロックは、グラフィックコマンドを含んでいる一つのパケットまたは複数のパケットを表す。「ユーザの入力」と「グラフィック」との両方の列において白いブロックによって表される時間間隔は、パケットが存在せず、従って全く活動がない。

30

【0099】

活動のない認識できる時間間隔を排除する方法の一実施形態は、以下の通りである。記録されたセッション内の第1のパケットが識別される。記録されたセッションは、表示データを表す複数のパケットを含む。記録されたセッション内の第1のパケットの直前のパケットは、第2のパケットとして識別される。第1の時間間隔が決定され、該時間間隔は、該第1のパケットと該第2のパケットとの間に生じる。第1の時間間隔が閾値を超えるという決定が行なわれる。記録されたセッション内のパケットのコンテンツが、該第1のパケットと該第2のパケットとの間の、第1の時間間隔よりも短い第2の時間間隔を有して、レンダリングされる。

40

【0100】

一実施形態において、プロトコルエンジンは決定を行なう。一部の実施形態において、プロトコルエンジンは、プレイバックデータ構造内に決定を格納する。一実施形態において、同じプロトコルエンジンが、プレイバックデータ構造に応答して、記録されたセッションをレンダリングする。別の実施形態において、決定を行なうプロトコルエンジンは、バックグラウンドプロトコルエンジンを含み、記録されたセッションをレンダリングするプロトコルエンジンは、フォアグラウンドプロトコルエンジンを含む。

【0101】

50

一実施形態において、時間間隔が閾値を超えるということを、プロトコルエンジンが決定するときには、プロトコルエンジンは、時間間隔を認識できる時間間隔として分類する。再生成された記録されたセッションのユーザが、活動のない期間が経過したことを視認し得る場合には、時間間隔は認識できる。一部の実施形態において、ポリシーは、閾値を決定する。他の実施形態において、プロトコルエンジンは、所定の閾値を用いてハードコード化される。この実施形態において、プロトコルエンジンは、プレイバックデータ構造内に命令を格納し、最初の時間間隔の代わりに、第1のバケットと第2のバケットとの間のより短い時間間隔をレンダリングする。別の実施形態において、時間間隔が閾値を超えるということを決定するプロトコルエンジンはまた、記録されたセッションのコンテンツをレンダリングする。この実施形態において、プロトコルエンジンは、より短い時間間隔をレンダリングする命令を、プレイバックデータ構造内に格納しない。認識できると分類されない時間間隔に関して、短くされた時間間隔は必要とされず、最初の時間間隔が、第1のバケットと第2のバケットとの間でレンダリングされる。

10

【0102】

再び図20を参照すると、流れ図は、記録されたセッションをレンダリングするプレイバック命令を生成するためにとられるステップの一実施形態を描いている。簡潔に概略すると、記録されたセッション内のバケットによって格納される入力の種類が識別され（ステップ2002）、入力の種類に応答して、バケットが、マーカー付けされる（ステップ2004）。次に、マーカー付けに応答して、バケットのコンテンツをレンダリングする宛先が、プレイバックデータ構造に格納される。

20

【0103】

一実施形態において、バケットによって格納された入力の種類は、バケットがレンダリングされるかどうかを決定する。一実施形態において、バケットはコンテンツを含んでいない。一部の実施形態において、少なくとも一つのバケットは、コンテンツを含んでいない。これらの実施形態において、コンテンツを含んでいない少なくとも一つのバケットを含む時間間隔が、識別される。これらの実施形態のうちの一部において、時間間隔は、レンダリングされない。

【0104】

一部の実施形態において、入力の種類は、キーボード、マウス、マイク、またはカメラを含むが、それらには限定されない一定の種類の入力デバイスからの入力を意味する。一実施形態において、入力の種類を識別するステップは、入力デバイスからの入力として、入力の種類を識別することをさらに包含する。別の実施形態において、入力の種類を識別するステップは、入力の種類をキーボード入力として識別することをさらに包含する。他の実施形態において、入力の種類は、入力デバイスに関連しない。これらの実施形態のうちの一つにおいて、入力の種類は、コマンドとして識別される。

30

【0105】

入力を含んでいるバケットは、バケットが含んでいる入力の種類に応答して、マーカー付けされる（ステップ2004）。一実施形態において、バケットは、ポリシーに応答してマーカー付けされる。この実施形態において、ポリシーは入力の種類を決定し、バケットがマーカー付けされる。別の実施形態において、マーカー付けは必要とされない。

40

【0106】

バケットをレンダリングする宛先が、マーカー付けに応答して、プレイバックデータ構造内に格納される（ステップ2006）。一部の実施形態において、宛先はバッファを含んでいる。一実施形態において、命令は、プレイバックデータ構造に格納され、バッファにバケットをレンダリングする。一実施形態において、バッファは、画面外のバッファであり、バケットのコンテンツが、バッファにレンダリングされるときには、コンテンツは、レンダリングのユーザにとって認識できない。一実施形態において、命令は、プレイバックデータ構造内に格納され、マーカー付けされたバケットのレンダリングを認識できる方法で、バッファに導く。

【0107】

50

一実施形態において、方法は、有意義な活動を含んでいない認識できる時間間隔を排除する。この実施形態において、ポリシーは、特定のタイプの入力を、有意義であるか、または無意味であるとして識別する。一部の実施形態において、ポリシーは、プロトコルエンジンにハードコード化され得る。他の実施形態において、アドミニストレータは、ポリシーを構成する。

【0108】

一部の実施形態において、パケットがコンテンツを含んでいない場合には、プロトコルエンジンは、パケットを無意味であるとして識別する。これらの実施形態のうちの一部において、パケットは、ユーザの活動が記録されたストリーム1714に記録されなかった時間間隔を表す。これらの実施形態において、プロトコルエンジンは、プレイバックデータ構造内に、いかなる無意味なパケットも認識できる方法ではレンダリングされない方法で、記録されたストリーム内の複数のパケットのそれぞれをレンダリングする宛先を格納する。図22は、記録されたストリームの再生の一実施形態を描き、該記録されたストリームのコンテンツは、プレイバックデータ構造に回答して、レンダリングされる。この実施形態において、プレイバックデータ構造に回答したレンダリングは、(図22における白いストライプ状のブロックによって記述されている)無意味な活動を含んでいる時間間隔の排除を可能にし、該無意味な活動は、まったく活動していない時間間隔を含む。このレンダリングは、有意義な記録されたセッションの再生をユーザに提供し、ポリシーは、いつコンテンツが有意義な活動を表示するかを決定する。一実施形態において、有意義な活動を表示しているコンテンツは、ユーザの入力のタイプを含む。

【0109】

一部の実施形態において、プロトコルエンジンは、証明可能に無意味な時間を含む前もって定義された時間に回答する入力タイプを識別する。一部の実施形態において、無意味な時間は、パケットがいかなるコンテンツも含んでいない時間間隔を含む。他の実施形態において、ポリシーは、入力タイプを定義し、該入力タイプは、無意味な時間を構成要素とする。さらに別の実施形態において、証明可能に無意味な時間を含む入力タイプの定義は、プロトコルエンジン内にハードコード化される。

【0110】

一部の実施形態において、パケットのコンテンツが、ユーザの活動を表示するが、ポリシーは、その活動を無意味な活動として識別する。これらの実施形態のうちの一つにおいて、ポリシーは、無意味な活動を、再生成された記録されたセッションのユーザにとって関心のないものとみなされる活動として定義する。これらの実施形態の別の実施形態において、ポリシーによって決定されるように、有意義なパケットは、再生成された記録されたセッションのユーザにとって関心のあるコンテンツを含んでいる。一実施形態において、無意味なパケットは、アプリケーションと有意義に相互作用する入力を表示するコンテンツを有していない。別の実施形態において、記録されたストリームが形成されるプロトコルデータストリームにおいて、アプリケーションデータを伝送するデバイスは、有意義なスクリーンのアップデートを伝送しない。

【0111】

一実施形態において、プロトコルエンジンは、記録されたセッション内の少なくとも一つのパケットに対して、パケットのコンテンツが、キーボード入力、マウス入力、またはコマンドメッセージを含むが、それらには限定されない入力のタイプを含んでいるかどうかを決定する。パケットが、キーボード入力のようなタイプの入力を含んでいる場合には、プロトコルエンジンは、パケットを有意義なパケットとしてマーカー付けする。パケットが、そのようなタイプの入力を含んでいない場合には、プロトコルエンジンは、そのパケットを無意味なパケットとしてマーカー付ける。一実施形態において、コンテンツの全てが、無意味な場合にのみ、パケットは無意味なものとなる。別の実施形態において、パケットは、一つより多いタイプの入力を含み、入力タイプのそれぞれは、意味があるものか、または無意味なものとしてマーカー付けされる。

【0112】

一実施形態において、プロトコルエンジンが、パケットを無意味なものとしてマーカー付けするときには、プロトコルエンジンは、パケットのコンテンツは認識できる方法で、レンダリングするべきではないということを決定する。その代わりに、一部の実施形態において、プロトコルエンジンは、パケットのコンテンツがバッファにレンダリングするべきであるということを決定する。これらの実施形態のうちの一つにおいて、バッファは、画面外バッファである。一実施形態において、パケットが、有意義なパケットとしてマーカー付けされた場合には、プロトコルエンジンは、パケットのコンテンツが、認識できる方法でレンダリングするべきであるということを決定する。一部の実施形態において、認識できる方法は、画面上でのレンダリングを含む。一実施形態において、プロトコルエンジンは、パケットが、認識できる方法で、バッファにレンダリングするべきであるということを決定する。この実施形態において、パケットのコンテンツは、ディスプレイの画面上と画面外のバッファとの両方でレンダリングする。プロトコルエンジンは、プレイバックデータ構造内に決定を格納する。

10

【0113】

図22に記述される一実施形態において、記録されたストリーム内の特定のパケットは、有意義なユーザの活動、この実施形態において、図22における黒いブロックによって表現されるマウスボタンの活動状態を示しているマウス入力表現しているコンテンツを有する。記録されたストリーム内の他のパケットは、図22におけるストライプ状のブロックによって表現されるマウスボタンの不活動状態を示しているマウス入力表現しているコンテンツを有する。プロトコルエンジンは、無意味な活動、例えば、マウスボタンの不活動状態を示しているマウス入力のみを含んでいる少なくとも一つのパケットを識別し、パケットのコンテンツが、認識できる方法で、レンダリングするべきではないという決定を、プレイバックデータ構造内に格納する。この決定を行なうことによって、プレイバックデータ構造に回答して、記録されたストリームのコンテンツをレンダリングするプロトコルエンジンは、再生成された記録されたセッションのユーザに関連する表示データのみを再生成し、その場合には、ポリシーが、関連性を定義しているか、またはプロトコルエンジンが、関連するコンテンツの定義を含んでいる。

20

【0114】

図23を参照すると、流れ図は、記録されたセッションをレンダリングする有意義な活動のない期間を排除するためにとられるステップの一実施形態を描いている。第1の時間間隔が決定され、該第1の時間間隔は、記録されたセッション内のマーカー付けされたパケットと直前にマーカー付けされたパケットとの間に発生する(ステップ2302)。記録されたセッションは、表示データを表示している複数のパケットを含んでいる。第1の時間間隔が閾値を超えているという決定が行なわれる(ステップ2304)。記録されたセッション内のパケットのコンテンツが、マーカー付けされたパケットと直前にマーカー付けされたパケットとの間の、第1の時間間隔よりも短い第2の時間間隔を有してレンダリングされる(ステップ2306)。

30

【0115】

一実施形態において、プロトコルエンジンは、決定を行なう。一部の実施形態において、プロトコルエンジンは、プレイバックデータ構造内に決定を格納する。一実施形態において、同じプロトコルエンジンが、プレイバックデータ構造に回答して、記録されたセッションをレンダリングする。別の実施形態において、決定を行なうプロトコルエンジンは、バックグラウンドプロトコルエンジンを含み、記録されたセッションをレンダリングするプロトコルエンジンは、フォアグラウンドプロトコルエンジンを含む。

40

【0116】

一部の実施形態において、パケットに含まれている入力のタイプに回答して、パケットが、有意義なパケットとしてマーカー付けされた後で、プロトコルエンジンは、第1の時間間隔の決定(ステップ2302)と、第1の時間間隔が閾値を超えているかどうかの決定(ステップ2304)とを行なう。これらの実施形態のうちの一つにおいて、パケットに含まれている出力のタイプは、パケットをマーカー付けするという決定に影響を与える

50

。一実施形態において、プロトコルエンジンは、有意義なものとしてマーカー付けされたパケットと直前の有意義なパケットの間の時間間隔を決定するか、または直前に有意義なパケットがない場合には、記録の開始を決定する。別の実施形態において、プロトコルエンジンは、マーカー付けされたパケットと直前のパケットとの間の第2の時間間隔を有する記録されたセッションのコンテンツをレンダリングする。該第2の時間間隔は、第1の時間間隔よりも短い時間間隔を含んでいる。別の実施形態において、プロトコルエンジンは、識別付けされたパケットと続いてマーカー付けされるパケットとの間の第2の時間間隔を有する記録されたセッションのコンテンツをレンダリングする。該第2の時間間隔は、第1の時間間隔よりも短い時間間隔を含んでいる。

【0117】

一実施形態において、プロトコルエンジンが、時間間隔が閾値を超えるということを決
定（ステップ2304）するときには、プロトコルエンジンは、その時間間隔を認識でき
る時間間隔として分類する。再生成された記録されたセッションのユーザが、活動のない
期間が経過しているということを理解し得る場合には、時間間隔は認識できる。一部の実
施形態において、ポリシーは、閾値を決定する。他の実施形態において、プロトコルエン
ジンは、所定の閾値でハードコード化される。この実施形態において、プロトコルエン
ジンは、プレイバックデータ構造内に命令を格納し、最初の時間間隔の代わりに、2つの有
意義なパケットとの間の短い時間間隔をレンダリングする。別の実施形態において、時間
間隔が閾値を超えるということを決するプロトコルエンジンはまた、記録されたセッシ
ョンのコンテンツをレンダリングする。この実施形態において、プロトコルエンジンは、
プレイバックデータ構造内で短い時間間隔をレンダリングする命令を格納しない。認識で
きると分類されていない時間間隔に関して、時間間隔は短くする必要はなく、最初の時間
間隔は、2つの有意義なパケットとの間でレンダリングされる。

【0118】

一部の実施形態において、記録されたストリーム内のパケットのコンテンツは、グラフ
ィックの更新を表し、スクリーン領域に影響を与える。一実施形態において、グラフィッ
クの更新は、点滅システムのトレイアイコン、タイトルバーまたはタスクバーエントリー
、ウェブページまたはウェブアプリケーション内の点滅テキスト、時計表示、システムア
ニメーション、アプリケーションアニメーション、および株価表示器、ならびに定期的
に更新される情報表示を含むが、それらには限定されない。一部の実施形態において、これ
らのようなグラフィックの更新は、記録されたストリームの再生成のユーザに対して無意
味なものであると決定される。これらの実施形態のうちの一つにおいて、プロトコルエン
ジンは、この決定を含んでいる。これらの実施形態の別のものにおいて、ポリシーは、少
なくとも一つのグラフィックの更新を無意味なものとして定義する。この実施形態にお
いて、アドミニストレータは、ポリシーを生成する。別の実施形態において、記録されたス
トリームの再生成のユーザはポリシーを生成する。

【0119】

図24を参照すると、流れ図は、記録されたセッションをレンダリングする際に、グラ
フィックの更新を排除するためにとられるステップの一実施形態を描いている。簡潔に概
略すると、グラフィックの更新が識別され（ステップ2402）、グラフィックの更新によ
って影響されるスクリーン領域が、決定される（ステップ2404）。影響を与えられ
るスクリーン領域の位置および状態の指標が格納され（ステップ2406）、スクリーン
領域の状態に影響を与えている第2のグラフィックの更新が、識別される（ステップ24
08）。次に、第2のグラフィックの更新が、第1のグラフィックの更新によって影響を
与えられる領域を変化させるかどうかに応答して、レンダリングの宛先、およびレンダ
リングの時間間隔が、示される（ステップ2410およびステップ2412）。

【0120】

一実施形態において、プロトコルエンジンは、図24に描かれたステップを実行する。
この実施形態において、プロトコルエンジンは、記録されたセッション内のパケットを識
別し、該記録されたセッションは、アプリケーションプログラムによって生成された表示

データを表示し、該パケットは、第1のグラフィックの更新を含んでいる。プロトコルエンジンは、第1のグラフィックの更新によって影響を与えられたスクリーン領域を決定する。一実施形態において、グラフィックの更新は、スクリーンのその部分に表示されるデータを変更することによって、スクリーン領域に影響を与える。プロトコルエンジンは、第1のグラフィックの更新、およびスクリーン領域の位置の後に、スクリーン領域の状態の表示を格納する。一実施形態において、プロトコルエンジンは、更新されたスクリーン領域のコピーを格納する。別の実施形態において、プロトコルエンジンは、更新されたスクリーン領域のハッシュを格納する。

【0121】

一実施形態において、プロトコルエンジンは、時間間隔内でスクリーン領域に影響を与える第2のグラフィックの更新を識別する。一部の実施形態において、ポリシーは、時間間隔の長さを決定する。これらの実施形態のうちの一つにおいて、ポリシーは、アプリケーションおよびオペレーティングシステムによって使用される人間スケールの循環期間のほぼ上限の時間間隔を決定する。一実施形態において、スクリーンの領域が循環表示を通過するときには、ユーザによって見られるように設計されている期間（例えば、一秒のほんの数分の一から最大で数秒）で、表示は人間スケールの循環期間を含んでいる。一部の実施形態において、プロトコルエンジンは、時間間隔の長さの定義を含んでいる。

【0122】

プロトコルエンジンが、第2のグラフィックの更新が第1のグラフィックの更新によって、影響を与えられているスクリーン領域に、影響を与えていることを識別する実施形態において、プロトコルエンジンは、第2のグラフィックの更新の後のスクリーン領域の状態が、第1のグラフィックの更新の後のスクリーン領域の状態から変化するかどうかを決定する。スクリーン領域が、第2のグラフィックの更新の後に変化しない場合には、第2のグラフィックの更新は、再生成された記録されたセッションにおいてレンダリングする必要がない。この実施形態におけるスクリーングラフィックの更新は、レンダリングする必要がない。なぜならば、プロトコルエンジンは、グラフィックの更新が、人間スケールの速度でコマンドを引き出す循環を実行するので、再生成された記録されたセッションのユーザにとって観察可能な（observable）更新を行うが、グラフィックの更新は、ユーザにとって無意味な情報を伝えることを決定するからである。一部の実施形態において、グラフィックの更新は、図形、カレット点滅、点滅するタスクバーのアイコン、ネットワーク活動インジケータ、またはスクロールテキストを含むが、それらには限定されないものによって、スクリーン領域に影響を与える。一部の実施形態において、ポリシーは、そのタイプのグラフィックの更新を用いて、スクリーン領域に影響を与えることが、有意義な活動を構成せず、ユーザに対して、記録されたセッションの再生成においてレンダリングするべきではないということを決定する。他の実施形態において、プロトコルエンジンは、この決定を含んでいる。

【0123】

一実施形態において、スクリーン領域に影響を与える第2のグラフィックの更新を含んでいる第2のパケットをレンダリングする宛先の指標は、第2のグラフィック更新の後に、スクリーン領域が変化するかどうかに応答して、プレイバックデータ構造内に格納される。別の実施形態において、スクリーン領域に影響を与える第2のグラフィックの更新を含んでいる第2のパケットと関連する、レンダリングする時間間隔の指標は、第2のグラフィックの更新の後のスクリーン領域の状態が、第1のグラフィックの更新の後のスクリーン領域の状態から変化するかどうかに応答して、プレイバックデータ構造内に格納される。

【0124】

図25は、第2のグラフィックの更新の後のスクリーン領域の状態が、第1のグラフィックの更新の後のスクリーン領域の状態から変化するかどうかに応答してプレイバックデータ構造内に格納される指標に応答して、再生成された記録されたセッションをレンダリングする一実施形態を描いている。一実施形態において、第1のグラフィックの更新によ

って影響を与えられるスクリーン領域は、第2のグラフィックの更新の後に変化しない。この実施形態において、指標は、第2のグラフィックの更新を認識できるようにレンダリングしないように、プレイバックデータ構造内に格納される。一実施形態において、第2のグラフィックの更新を認識できるようにレンダリングしないことは、画面上ではなく、画面外において第2のグラフィックの更新をレンダリングすることを含んでいる。一部の実施形態において、第2のグラフィックの更新を認識できるようにレンダリングしないことは、第2のグラフィックの更新を画面外のバッファにレンダリングすることを含んでいる。一実施形態において、第2のグラフィックの更新を認識できるようにレンダリングしないことは、第2のグラフィックの更新をレンダリングしないことを含んでいる。一部の実施形態において、第2のグラフィックの更新を認識できるようにレンダリングしない決定は、グラフィックの更新がレンダリングされない認識できる指標をレンダリングすることを含んでいる。これらの実施形態のうちの一つにおいて、再生成された記録されたセッションのユーザは、第2のグラフィックの更新が認識できるようにレンダリングすることを要求し得る。

10

【0125】

図25は、第2のグラフィックの更新の後のスクリーン領域の状態が、第1のグラフィックの更新の後のスクリーン領域の状態から変化しないということを決定し、かつ循環の更新の検出に応答して、第2のグラフィックの更新をレンダリングしないことを決定することによって、循環の更新が検出される実施形態を描いている。第2のグラフィックの更新によって影響を与えられるスクリーン領域の状態が、第1のグラフィックの更新の後のスクリーン領域の状態から変化する一実施形態において、パケットのコンテンツを、認識できる方法で、バッファにレンダリングする決定が行なわれる。

20

【0126】

一部の実施形態において、複数のパケットのコンテンツが、グラフィックの更新を表示する。これらの実施形態のうちの一つにおいて、認識できる方法でグラフィックの更新をレンダリングする決定は、スクリーン領域上の二つよりも多いグラフィックの更新の効果に応答して行なわれる。一実施形態において、グラフィックの更新をレンダリングする宛先の決定は、識別された複数のパケットのうちの各パケットのコンテンツによって表示されるグラフィックの更新に応答する。

【0127】

一部の実施形態において、記録されたストリーム内のパケットのコンテンツは、相互作用シーケンスを表示する。一実施形態において、相互作用シーケンスは、ログオンシーケンス、ログオフシーケンス、または認証の入力を含むが、それらには限定されない。一部の実施形態において、これらのような相互作用シーケンスは、記録されたストリームの再生成のユーザにとって無意味なものであるとして決定される。これらの実施形態のうちの一つにおいて、プロトコルエンジンは、この決定を含んでいる。これらの実施形態のうちの別の実施形態において、ポリシーは、少なくとも一つの相互作用を無意味であるとして定義する。この実施形態において、アドミニストレータは、ポリシーを生成する。別の実施形態において、記録されたストリームの再生成のユーザは、ポリシーを生成する。

30

【0128】

図26を参照すると、流れ図は、記録されたセッションをレンダリングする相互作用シーケンスを排除するためにとられるステップの一実施形態を描いている。少なくとも1つのパケットの相互作用シーケンスの開始が、ポリシーに응答して識別される(ステップ2602)。プレイバックデータ構造において、相互作用シーケンスがバッファにレンダリングすべき指標が存在する(ステップ2604)。少なくとも一つのパケットの相互作用シーケンスの終了が識別される(ステップ2606)。相互作用シーケンスの開始が識別されるのに先行するパケットと、相互作用シーケンスの終了が識別されるのに続くパケットとの間の第1の時間間隔が、識別される(ステップ2608)。プレイバックデータ構造は、第1の時間間隔よりも短い第2の時間間隔をレンダリングする指標を含んでいる(ステップ2610)。

40

50

【0129】

一実施形態において、プロトコルエンジンは、相互作用シーケンスを排除する識別および表示を行なう。相互作用シーケンスの開始の識別が、行なわれる（ステップ2602）。一実施形態において、相互作用シーケンスの開始は、視覚的なマーカーを識別することによって識別される。一実施形態において、視覚的なマーカーは、全てのセッションに対して同じ方法で表示される認証ウインドウを含んでいる。別の実施形態において、視覚的なマーカーは、ブランクスクリーン、および次に、デスクトップバックグラウンドによる認証ウインドウの移動を含んでいる。一実施形態において、視覚的なマーカーは、識別可能なアイコンの表示を含んでいる。

【0130】

一部の実施形態において、相互作用シーケンスの開始は、相互作用シーケンスの開始時間を決定することによって識別される。これらの実施形態のうちの一つにおいて、コンポーネントが、相互作用シーケンスにおけるイベントの開始時間を検出する。これらの実施形態の別の実施形態において、コンポーネントは、ログオンシーケンスの開始時間を検出する。これらの実施形態のさらに他の実施形態において、コンポーネントは、ログオフシーケンスの開始時間を検出する。一実施形態において、相互作用シーケンスの開始の識別は、入力フォーカスを用いてウインドウを識別することに対応する。

【0131】

プレイバックデータ構造において、相互作用シーケンスがバッファ内でレンダリングすべきであるという指示が行なわれる（ステップ2604）。識別された相互作用シーケンスが、認識できるようにレンダリングすべきではない実施形態において、相互作用シーケンスは、バッファにレンダリングされる。相互作用シーケンスをバッファにレンダリングすることは、相互作用シーケンスがレンダリングのユーザにとって認識できなくなることとなる。ポリシーまたはユーザが、相互作用シーケンスを無意味であるとして分類する実施形態に関して、このレンダリングは、無意味な相互作用シーケンスを排除することとなる。

【0132】

相互作用シーケンスの終了の識別がまた行なわれる（ステップ2606）。一部の実施形態において、相互作用シーケンスの終了は、視覚的なマーカーを識別することによって識別される。他の実施形態において、相互作用シーケンスの終了は、相互作用シーケンスの終了時間を決定することによって識別される。これらの実施形態のうちの一つにおいて、コンポーネントは、相互作用シーケンスにおけるイベントの終了時間を検出する。これらの実施形態のうちの一つの別の実施形態において、コンポーネントは、ログオンシーケンスの終了時間を検出する。これらの実施形態のうちの一つのさらに別の実施形態において、コンポーネントは、ログオフシーケンスの終了時間を検出する。別の実施形態において、相互作用シーケンスの終了を識別することは、入力フォーカスを用いたウインドウを識別することに対応する。

【0133】

一部の実施形態において、相互作用シーケンスは、アプリケーションの使用を含んでいる。これらの実施形態のうちの一つにおいて、ポリシーは、認識できる方法においてレンダリングすべきではないアプリケーションの使用を含んでいる相互作用シーケンスを識別する。一実施形態において、そのようなアプリケーションは、文書処理ドキュメントを含むが、それらには限定されない。

【0134】

これらの実施形態のうちの一つにおいて、相互作用シーケンスの開始は、入力フォーカスを有するアプリケーションを識別することによって識別される。パケットのコンテンツがフォーカスを有するウインドウを表示するときには、ウインドウを作成した処理に責任のあるアプリケーションに関して、決定が行なわれる。一実施形態において、フォーカスを有するウインドウを表示しているパケットのコンテンツは、入力フォーカスにおける変化を示しているウインドウ通知メッセージを含む。責任のあるアプリケーションが、認識

10

20

30

40

50

できるようにレンダリングするべきではない相互作用シーケンスの開始を識別する場合には、指標は、プレイバックデータ構造に格納され、相互作用シーケンスをバッファにレンダリングする。相互作用シーケンスの終了が、相互作用シーケンスのアプリケーションと関連しない処理によって支配されるウィンドウによる、フォーカスの獲得を識別することによって識別される。

【0135】

一実施形態において、第1の時間間隔は、相互作用シーケンスと関連付けられる。相互作用シーケンス自体がレンダリングしない実施形態において、相互作用シーケンスと関連する時間間隔を認識できるようにレンダリングすることは、レンダリングのユーザにとって認識できる期間となり、該レンダリングにおいて、表示データはレンダリングせず、ユーザは、相互作用シーケンスの後のパケットのコンテンツをレンダリングする前の時間間隔を通して待機する。一実施形態は、最初の時間間隔の代わりに、短い時間間隔をレンダリングすることによって、相互作用シーケンスと関連する時間間隔を排除する。この実施形態において、相互作用シーケンスの開始の識別に先行するパケットと、相互作用シーケンスの終了の識別に続くパケットとの間の第1の時間間隔が、識別される（ステップ2608）。プレイバックデータ構造は、第1の時間間隔よりも短い第2の時間間隔をレンダリングする指標を含んでいる（ステップ2610）。

【0136】

一実施形態において、プロトコルエンジンは、記録されたセッション内のパケットのコンテンツをレンダリングし、ユーザに記録されたセッションの再生を提供する。これらの実施形態のうちの一部において、プロトコルエンジンは、少なくとも一つのパケットのコンテンツをレンダリングする間の時間間隔を自動的に変化させ、状況依存（context-sensitive）のタイムワーププレイバックとなる。これらの実施形態において、レンダリングは、ユーザに表示される表示データを理解するユーザの能力を見積もる。一実施形態において、パケットのコンテンツによって表示される表示データが、ポリシーによって定義されるように、複雑さ、または重要性の程度が増加したことを、プロトコルエンジンが決定するときには、パケットのコンテンツをレンダリングする間の時間間隔が増加する。別の実施形態において、パケットのコンテンツによって表示される表示データが、ポリシーによって定義されるように、複雑さ、または重要性の程度が減少したことを、プロトコルエンジンが決定するときには、パケットのコンテンツをレンダリングする間の時間間隔が減少する。これらの実施形態において、プロトコルエンジンは、表示データを理解するユーザの能力を見積もり、コンテンツをゆっくりとレンダリングし、ユーザにレンダリングを理解する時間を与えるか、またはユーザが理解する時間をあまり必要としないときには、コンテンツを速くレンダリングするかのいずれかである。

【0137】

図27を参照すると、流れ図は、記録されたコンピュータセッションをレンダリングする際の自動的なタイムワーププレイバックにおいてとられるステップの一実施形態を描いている。プロトコルエンジンは、記録されたセッションを受信し（ステップ2702）、記録されたセッションは、複数のパケットを含み、表示データを表示し、プロトコルエンジンは、記録されたセッション内の複数のパケットのうちの少なくとも一部によって表現される複雑さの程度を決定する（ステップ2704）。プロトコルエンジンは、記録されたセッション内の複数のパケットのうちの少なくとも一部の間の時間間隔を識別し（ステップ2706）、記録されたセッション内の複数のパケットのうちの少なくとも一部によって表示される複雑さの程度に回答して、時間間隔を改変する（ステップ2708）。プロトコルエンジンは、プレイバックデータ構造内に、改変された時間間隔を格納し（ステップ2710）、記録されたデータストリームは、プレイバックデータ構造に回答してレンダリングされる（ステップ2712）。

【0138】

一実施形態において、複雑さの程度を決定し、時間間隔を識別し、時間間隔を改変し、改変を格納するプロトコルエンジンは、バックグラウンドプロトコルエンジンである

。これらの実施形態のうちの一つにおいて、バックグラウンドプロトコルエンジンはまた、記録されたストリームをレンダリングする。これらの実施形態のうちの一つの実施形態において、フォアグラウンドプロトコルエンジンは、プレイバックデータ構造にตอบสนองして、記録されたストリームをレンダリングする。一部の実施形態において、バックグラウンドプロトコルエンジンおよびフォアグラウンドプロトコルエンジンは、同じデバイスに存在する。他の実施形態において、バックグラウンドプロトコルエンジンおよびフォアグラウンドプロトコルエンジンは、別のデバイスに存在する。

【0139】

一部の実施形態において、プロトコルエンジンは、記録されたセッション内の複数のパケットの少なくとも一部によって表示される複雑さの程度を決定する（ステップ2704）。これらの実施形態のうちの一つにおいて、プロトコルエンジンは、キーボード入力における適切なシーケンスのタイピングを識別することによって、複雑さの程度を決定する。一実施形態において、プロトコルエンジンは、キーボード入力における適切なシーケンスのタイピングを識別するために含まれるキーの少なくとも一つのタイピングを調べる。別の実施形態において、プロトコルエンジンは、キーボード入力における適切なシーケンスのタイプの発見的な見積もりを完成させるためにレンダリングされる少なくとも一つのグリフのシーケンスを検査する。

【0140】

これらの実施形態のうちの一つにおいて、プロトコルエンジンは、キーの特徴によって決定されたキーの分類を格納する。キーの特徴は、プリント可能であるか、またはプリント可能ではない文字、白いスペース、ナビゲーションキー、またはファンクションキーを含むが、それらには限定されず、それらの特徴の組み合わせを含む。一実施形態において、プロトコルエンジンは、プリント可能である文字と予備のナビゲーションキーとを含んでいる入力セクションが、通常のタイピングを構成し、ほとんど不可視であるキーを用いるセクションは、通常のタイピングを構成しないということを決定する。一実施形態において、プロトコルエンジンは、識別された白いスペースの量にตอบสนองして、複雑さの程度を決定する。この実施形態において、プロトコルエンジンは、白いスペースキーが、一般的なタイピングのパターンにおいて、平均して、大体5～8文字ごとに現れることを示している文書処理の定義を含んでいる。

【0141】

一実施形態において、プロトコルエンジンは、プリント可能ではない文字の形状を使用して、複雑さの程度を決定する。別の実施形態において、プロトコルエンジンは、キーストロークシーケンスにアクセスし、適宜、近接して現れる白ではないスペースのプリント可能である文字のシーケンスを識別する。この実施形態において、プロトコルエンジンは、キーストロークシーケンスを辞書と比較し、妥当な単語を識別し、妥当ではない単語に対する妥当な単語を、ユーザが理解する能力に対する複雑さの程度を決定するキャパシティを含む。

【0142】

別の実施形態において、プロトコルエンジンは、パケットのコンテンツが、グリフをレンダリングするコマンドを含んでいるということを決定する。この実施形態において、プロトコルエンジンは、グリフを使用し、表示データがユーザのタイピング活動を表すかどうかを決定する。この実施形態において、グリフのレンダリング率が、わずかな遅れを有するキーボード入力率を見積もる場合には、キーストロークが直接的にグリフとなる可能性が高く、従ってユーザがタイピングしている可能性が極めて高い。一実施形態において、プロトコルエンジンは、生成されたグリフを用いて入力されたキーに相関する。別の実施形態において、プロトコルエンジンは、レンダリングされたグリフの空間的なシーケンス（左から右、右から左など）を決定し、ユーザがタイピングしているということを決定する。一実施形態において、プロトコルエンジンは、複数のパケットのコンテンツを分析し、コンテンツによって表されたパターンおよび活動を識別した結果にตอบสนองして、複雑さの程度の決定を行なう。

【 0 1 4 3 】

他の実施形態において、プロトコルエンジンは、マウス入力のタイプの識別にตอบสนองして、複雑さの程度の決定を行なう。一実施形態において、特に、クリックがタイピングシーケンスに続く場合には、プロトコルエンジンは、マウスのクリックを表しているマウス入力、理解するために、ゆっくりとしたレンダリング率を必要とし得る活動をもたらすということを決定する。別の実施形態において、プロトコルエンジンは、マウスのクリックを表さないマウス入力、ユーザが表示データを理解する能力に影響を与えず、従って複雑さの程度に影響を与えないということを決定する。

【 0 1 4 4 】

他の実施形態において、プロトコルエンジンは、グラフィックの更新の複雑さの発見的な見積もりを識別することに対応して、複雑さの程度の決定を行なう。一実施形態において、限定されるものではないが、更新される領域のサイズ、グラフィックコマンドによって変更される領域の面積サイズ、個々の領域に対する更新の頻度の履歴、循環グラフィックコマンド、グラフィックコマンドの数、グラフィックコマンドの頻度、コンテンツがグラフィックコマンドを含んでいる隣接するパケットの間の時間間隔、またはグラフィックの更新のタイプに基づいて、プロトコルエンジンは、グラフィックの更新の複雑さの発見的な見積もりを識別する。プロトコルエンジンが、グラフィックの更新に関する複雑さの程度が低いと識別する実施形態において、プロトコルエンジンは、グラフィックの更新を含んでいるパケットによって表される複雑さの程度が低いと決定する。プロトコルエンジンが、グラフィックの更新に関して複雑さの程度が高いと識別する実施形態において、プロトコルエンジンは、グラフィックの更新を含んでいるパケットによって表される複雑さの程度が高いと決定する。

【 0 1 4 5 】

一実施形態において、プロトコルエンジンは、記録されたセッション内の複数のパケットのうちの少なくとも一部の間の時間間隔を識別する（ステップ 2706）。この実施形態において、プロトコルエンジンは、複雑さの程度の決定に対応して、時間間隔を改変する（ステップ 2708）。記録されたセッション内の複数のパケットのうちの少なくとも一部が、程度の高い複雑さと関連する表示データを表しているコンテンツを有する実施形態において、プロトコルエンジンは、パケットの間の時間間隔を増加させ、ユーザがレンダリング時間の増加によりレンダリングされた表示データを理解することを可能にする。記録されたセッション内の複数のパケットのうちの少なくとも一部が、程度の低い複雑さと関連する表示データを表しているコンテンツを有する別の実施形態において、プロトコルエンジンは、ユーザがレンダリングされた表示データを理解するために必要とする時間量が減少したことを反映するように、パケットの間の時間間隔を減少させる。一実施形態において、ユーザは、プロトコルエンジンによってレンダリングされた時間量とは異なる、レンダリングされたパケットのコンテンツの間の時間量を必要とする。この実施形態において、ユーザは、レンダリングされた表示データをユーザが理解するために必要とする時間量を反映するように、時間間隔を改変する。一部の実施形態において、プロトコルエンジンはまた、複数のパケットのうちの少なくとも一部と、複数のパケットのうちの他のパケットとの間の時間間隔を識別し、パケットのそれらのセットの間で識別された時間間隔を改変する。

【 0 1 4 6 】

一部の実施形態において、プロトコルエンジンは、記録されたセッション内のパケットと関連する第 1 のマーカーを識別する。一実施形態において、パケットはマーカーを含む。別の実施形態において、記録されたセッションはマーカーを含む。

【 0 1 4 7 】

一実施形態において、表示データのレンダリングのユーザは、マーカーを定義する。別の実施形態において、プロトコルエンジンは、マーカーを定義する。プロトコルエンジンがマーカーを識別する実施形態において、プロトコルエンジンは、第 1 のマーカーに対応して、時間間隔を改変する。一実施形態において、プロトコルエンジンは、時間間隔を増

加させ、表示データのレンダリングのユーザに、第1のマーカと関連するパケットのコンテンツを理解する追加の時間を提供する。別の実施形態において、プロトコルエンジンは、第2のパケット内の第2のマーカを識別する。この実施形態において、プロトコルエンジンは、第1のマーカと第2のマーカとの間の距離に回答して、時間間隔を改変する。この実施形態において、プロトコルエンジンは、マーカ付けされたパケットのコンテンツによって表される表示データの理解のために時間を増加させ、マーカ付けされていないパケットのコンテンツによって表されるデータの理解のために時間を減少させる。一実施形態において、ユーザは、ユーザにとって関心のある表示データのためにマーカを定義し、プロトコルエンジンは、ユーザにとって関心ある表示データのための追加の時間をレンダリングし、マーカによって決定されるように、ユーザにとって関心のない表示データのためのレンダリング時間減少させる。

【0148】

一実施形態において、プロトコルエンジンは、記録されたセッション内の複数のパケットのうちの少なくとも一部における第1のマーカを識別し、該マーカは、記録されたセッション内の複数のパケットのうちの少なくとも一部における最初のパケットを指している。プロトコルエンジンは、第1のマーカに回答して時間間隔を改変する。プロトコルエンジンは、記録されたセッション内の複数のパケットのうちの少なくとも一部における第2のパケットにおける第2のマーカを識別し、該第2のマーカは、記録されたセッション内の複数のパケットのうちの少なくとも一部における最後のパケットを指し、第1のマーカと第2のマーカとの間の時間間隔に回答して、時間間隔を改変する。

【0149】

一実施形態において、プロトコルエンジンは、プレイバックデータ構造内に改変された時間間隔を格納し(ステップ2710)、記録されたストリームは、プレイバックデータ構造のコンテンツに回答してレンダリングされる(ステップ2712)。一実施形態において、プロトコルエンジンはまた、改変された時間間隔に関するプレイバックデータ構造の命令に回答して、記録されたストリームをレンダリングする。別の実施形態において、別個のフォアグラウンドプロトコルエンジンが、記録されたストリームをレンダリングする。

【0150】

一部の実施形態において、アプリケーションを用いた、記録された相互作用は、ユーザが、レンダリングされた表示データの理解のために時間を増加させるために、レンダリングするための時間を増加させることを必要とするという決定が行なわれる。これらの実施形態のうちの一部において、時間を増加させることを必要とするアプリケーションは、時間を増加させることを必要としないアプリケーションよりも、重要なアプリケーションを含むという決定が行なわれる。これらの実施形態のうちの一つにおいて、ユーザが決定を行なう。これらの実施形態のうちの別の実施形態において、ポリシーが決定を行なう。これらの実施形態のうちのさらに別の実施形態において、プロトコルエンジンは、時間の増加を必要とするアプリケーションの定義を含む。

【0151】

図28を参照すると、流れ図は、記録されたコンピュータセッション内の識別されたアプリケーションに回答して、自動的なタイムワーププレイバックするためにとられるステップの一実施形態を描いている。複数のパケットを含み、表示データを表している記録されたセッションが受信される(ステップ2802)。フォーカスを有するウィンドウを表しているコンテンツを有する第1のパケットが、識別され、該ウィンドウは、アプリケーションを指している(ステップ2804)。第1のパケットのコンテンツのレンダリングに先立ち、コンテンツがレンダリングする第2のパケットと、第1のパケットのコンテンツのレンダリングの後に、コンテンツがレンダリングする第3のパケットとの間で、時間間隔は識別される(ステップ2806)。識別された時間間隔は、示されたアプリケーションに回答して改変される(ステップ2808)。記録されたストリーム内の少なくとも一つのパケットは、改変に回答してレンダリングされる(ステップ2810)。

【 0 1 5 2 】

一実施形態において、プロトコルエンジンは、記録されたセッションを受信する（ステップ 2 8 0 2）。この実施形態において、プロトコルエンジンはまた、フォーカスを有するウィンドウを表しているコンテンツを有する第 1 のパケットを識別し、該ウィンドウはアプリケーションを示している（ステップ 2 8 0 4）。一実施形態において、フォーカスを有するウィンドウを表しているパケットのコンテンツは、入力フォーカスの変更を示しているウィンドウ通知メッセージを含んでいる。一実施形態において、第 1 のパケットのコンテンツのレンダリングに先立ち、コンテンツがレンダリングする第 2 のパケットと、第 1 のパケットのコンテンツのレンダリングの後に、コンテンツがレンダリングする第 3 のパケットとの間で、時間間隔は識別される（ステップ 2 8 0 6）。この実施形態において、フォーカスを有するアプリケーションウィンドウを表しているコンテンツをレンダリングするのに先立ち、コンテンツが、レンダリングするパケットと、コンテンツが、フォーカスを有するアプリケーションウィンドウを表しているパケットと、コンテンツが、もはやフォーカスを有していないアプリケーションウィンドウを表しているパケットとを、プロトコルエンジンは識別する。

10

【 0 1 5 3 】

一部の実施形態において、プロトコルエンジンは、フォーカスを有するアプリケーションに先立つ時間間隔を改変する。他の実施形態において、プロトコルエンジンは、フォーカスを有するアプリケーションに続く時間間隔を改変する。一実施形態において、プロトコルエンジンは時間間隔を決定し、その時間間隔において、アプリケーションウィンドウはフォーカスを有し、アプリケーションのタイプに応答して、その時間間隔を改変する。一実施形態において、プロトコルエンジンは、識別された時間間隔を増加させる。この実施形態において、プロトコルエンジンは、レンダリングのユーザのアプリケーションをレビューする時間の量を増加させる。別の実施形態において、プロトコルエンジンは、識別された時間間隔を減少させる。この実施形態において、プロトコルエンジンは、レンダリングのユーザのアプリケーションをレビューする時間の量を減少させ、アプリケーションにおける関心の量が減少したことを反映する。

20

【 0 1 5 4 】

一実施形態において、プロトコルエンジンは、改変に応答して、記録されたストリーム内の少なくとも一つのパケットをレンダリングする。一実施形態において、プロトコルエンジンは、記録されたストリーム内の少なくとも一つのパケットのコンテンツを、バッファにレンダリングする。一実施形態において、バッファにレンダリングすることは、パケットのコンテンツを認識できる方法ではレンダリングしない。別の実施形態において、プロトコルエンジンは、記録されたストリーム内の少なくとも一つのパケットのコンテンツをバッファに、認識できる方法でレンダリングする。一部の実施形態において、プロトコルエンジンは、プレイバックデータ構造内の改変された時間間隔を示し、別個のプロトコルエンジンは、プレイバックデータ構造内に格納される情報に応答して、記録されたセッションをレンダリングする。

30

【 0 1 5 5 】

図 2 9 を参照すると、ブロック図は、記録されたコンピュータセッションをレンダリングする際の自動的なタイムワーププレイバックに関するシステムの一実施形態を描き、プロトコルエンジン 2 9 0 2、記録されたストリーム 2 9 1 0、プレイバックデータ構造 2 9 0 4、およびディスプレイ 2 9 0 8 を含んでいる。簡潔に概略すると、プロトコルエンジン 2 9 0 2 は、記録されたストリーム 2 9 1 0 を受信することに応答して、プレイバックデータ構造 2 9 0 4 を生成し、該記録されたストリーム 2 9 1 0 は、複数のパケットを含み、該プロトコルエンジン 2 9 0 2 は、生成されたプレイバックデータ構造 2 9 0 4 に応答して、記録されたデータストリーム内の少なくとも一つのパケットをレンダリングする。

40

【 0 1 5 6 】

一実施形態において、プロトコルエンジン 2 9 0 2 は、バックグラウンドプロトコルエ

50

ンジンとフォアグラウンドプロトコルエンジンとを含む。この実施形態において、バックグラウンドプロトコルエンジンは、記録されたストリーム 2910 を受信し、プレイバックデータ構造 2904 を生成する。この実施形態において、フォアグラウンドプロトコルエンジンは、記録されたストリーム 2910 を受信し、生成されたプレイバックデータ構造 2904 に応答して、記録されたストリーム内の少なくとも一つのバケットをレンダリングする。一実施形態において、バックグラウンドプロトコルエンジンとフォアグラウンドプロトコルエンジンとは、同じデバイスに存在し得る。別の実施形態において、バックグラウンドプロトコルエンジンは、第 1 のデバイスに存在し、フォアグラウンドプロトコルエンジンは、第 2 のデバイスに存在する。

【0157】

10

別の実施形態において、システムは、プレイバックデータ構造 2904 を生成し、生成されたプレイバックデータ構造 2904 に応答して、記録されたストリーム内の少なくとも一つのバケットをレンダリングする単一のプロトコルエンジン 2902 を備える。

【0158】

一実施形態において、プロトコルエンジン 2902 は、記録されたセッションをレンダリングするために、少なくとも一つの命令をプレイバックデータ構造内に格納する。一実施形態において、命令は、記録されたセッション内のバケットのコンテンツをレンダリングするための識別された時間間隔の改変を含む。別の実施形態において、プロトコルエンジンは、プレイバックデータ構造内にメタデータをプレイバックデータ構造内に格納する。この実施形態において、メタデータは、バケットのコンテンツをレンダリングするための優先順位の高い命令を含む。

20

【0159】

一実施形態において、プロトコルエンジンは、プレイバックデータ構造のコンテンツに応答して、記録されたセッション内の少なくとも一つのバケットのコンテンツをレンダリングする。一実施形態において、プロトコルエンジンは、記録されたセッション内の少なくとも一つのバケットを、認識できる方法でバッファにレンダリングする。別の実施形態において、プロトコルエンジンは、記録されたセッション内の少なくとも一つのバケットをバッファにレンダリングする。

【0160】

一部の実施形態において、バケットのレンダリングされたコンテンツは、最初の表示データの再生の簡素化を提供する。他の実施形態において、バケットのレンダリングされたコンテンツは、表示データのカスタム化されたバージョンを提供する。一実施形態において、認識できる方法でバケットのコンテンツをレンダリングする決定は、ポリシーまたはユーザの要求に応答する。これらの実施形態は、記録されたセッションのレンダリングに対する制御を、ユーザに提供する。

30

【0161】

本発明は、一つ以上の工業製品に組み込まれる一つ以上のコンピュータで読取可能であるプログラムとして提供され得る。工業製品は、フロッピー（登録商標）ディスク、ハードディスク、コンパクトディスク、デジタル多目的ディスク、フラッシュメモリカード、PROM、RAM、ROM、または磁気テープであり得る。概して、コンピュータで読取可能なプログラムは、任意のプログラミング言語で実装され得る。使用され得る言語の一部の例は、C、C++、C#またはJAV A（登録商標）を含む。ソフトウェアプログラムは、オブジェクトコードとして、一つ以上の工業製品に格納され得る。

40

【0162】

本発明は、特定の好適な実施形態を参照して示され、記述されてきたが、形状および詳細に関する様々な変更が、特許請求の範囲によって定義されるように、本発明の意図および範囲を逸脱することなく行われ得ることを当業者は理解するべきである。

【図面の簡単な説明】

【0163】

【図1A】図1Aは、本発明の一実施形態を実施するために最適なクライアントサーバシ

50

システムを描写するブロック図である。

【図 1 B】図 1 B は、本発明と関連して有用なコンピュータの実施形態を描写するブロック図である。

【図 1 C】図 1 C は、本発明と関連して有用なコンピュータの実施形態を描写するブロック図である。

【図 2】図 2 は、本発明が行われ得る、ネットワーク 2 0 0 の実施形態を描写するブロック図である。

【図 3】図 3 は、ネットワーク 2 0 0 上のレコーダに対する配置の代替的な実施形態を描写するブロック図である。

【図 4】図 4 は、ネットワーク 2 0 0 上のレコーダに対する配置の代替的な実施形態を描写するブロック図である。

【図 5】図 5 は、プロトコルデータストリームによって表されたディスプレイデータを再生成するシステムの一実施形態を描写するブロック図である。

【図 6】図 6 は、サーバが生成したデータの記録およびプレイバックの方法を描写する流れ図である。

【図 7】図 7 は、プロトコルデータストリームによって表されるディスプレイデータを記録するシステムにおけるレコーダをさらに詳細に描写するブロック図である。

【図 8】図 8 は、プロトコルデータストリームの記録の一実施形態を描写する。

【図 9】図 9 は、ステートフル遠隔提示プロトコルのプレイバックの間の、リアルタイムシーキングの方法において取られるステップの一実施形態を描写する、流れ図である。

【図 1 0】図 1 0 は、遠隔提示プロトコルのプレイバックの間に、リアルタイムシーキングを可能にするステートスナップショットを生成するために取られるステップの一実施形態を描写する、流れ図である。

【図 1 1】図 1 1 は、ステートフル遠隔提示プロトコルのプレイバックの間のリアルタイムシーキングのシステムを描写するブロック図である。

【図 1 2】図 1 2 は、ステートスナップショットの適応性のある生成のために取られるステップの一実施形態を描写する流れ図である。

【図 1 3】図 1 3 は、一実施形態の、三種類のシーク確率分布を描写するダイアグラムである。

【図 1 4】図 1 4 は、決定されたシーク確立分布に応答してステートスナップショットを生成する一実施形態を描写するダイアグラムである。

【図 1 5】図 1 5 は、プロトコルデータストリームの提示の、ユーザに対する使用パターンの一実施形態を描写する。

【図 1 6】図 1 6 は、ステートスナップショットの適応的な生成のためのシステムの一実施形態を描写するブロック図であって、バックグラウンドプロトコルエンジン、フォアグラウンドプロトコルエンジン、プロトコルデータストリーム、活動プロファイルおよびステートスナップショットを含む図である。

【図 1 7】図 1 7 は、記録されたセッションをレンダリングするシステムの一実施形態を描写するブロック図である。

【図 1 8】図 1 8 は、記録されたコンピュータセッションのプレイバックに対するプレイバック命令を生成するために取られるステップの一実施形態を描写する流れ図である。

【図 1 9】図 1 9 は、記録されたコンピュータセッションのプレイバックの方法において取られるステップの一実施形態を描写する流れ図である。

【図 2 0】図 2 0 は、記録されたセッションをレンダリングするためのプレイバック命令を生成するために取られるステップの一実施形態を描写する流れ図である。

【図 2 1】図 2 1 は、再生成された記録されたストリームの一実施形態を描写し、そのコンテンツは、プレイバックデータ構造に応答してレンダリングされる。

【図 2 2】図 2 2 は、意味のあるユーザ活動を表すコンテンツを有するストリームを記録する際の、特定のパケットの一実施形態を描写し、この実施形態において、マウス入力は、アクティブマウスボタンの状態を示す。

10

20

30

40

50

【図 2 3】図 2 3 は、記録されたセッションをレンダリングする際に意味のない活動を伴う期間を消去するために取られるステップの一実施形態を描写する流れ図である。

【図 2 4】図 2 4 は、記録されたセッションをレンダリングする際にグラフィックの更新を消去するために取られるステップの一実施形態を描写する流れ図である。

【図 2 5】図 2 5 は、再生成された記録されたセッションのレンダリングの一実施形態を描写し、第 2 のグラフィックの更新の後のスクリーン領域の状態が、第 1 のグラフィックの更新の後のスクリーン領域の状態と変わるかどうかに応答する。

【図 2 6】図 2 6 は、記録されたセッションをレンダリングする際に相互作用シーケンスを消去するために取られるステップの一実施形態を描写する流れ図である。

【図 2 7】図 2 7 は、記録されたコンピュータセッションをレンダリングする際に、自動タイムワーププレイバックにおいて取られるステップの一実施形態を描写する流れ図である。

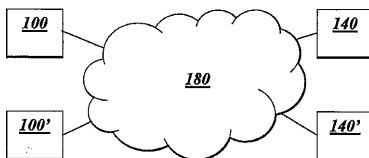
【図 2 8】図 2 8 は、記録されたコンピュータセッションをレンダリングする際の識別されたアプリケーションに応答する、自動タイムワーププレイバックに対して取られるステップの一実施形態を描写する流れ図である。

【図 2 9】図 2 9 は、記録されたコンピュータセッションをレンダリングする際の、自動タイムワーププレイバックのシステムの一実施形態を描写するブロック図である。

10

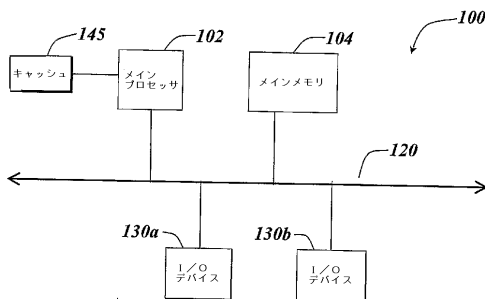
【図 1 A】

FIG. 1A



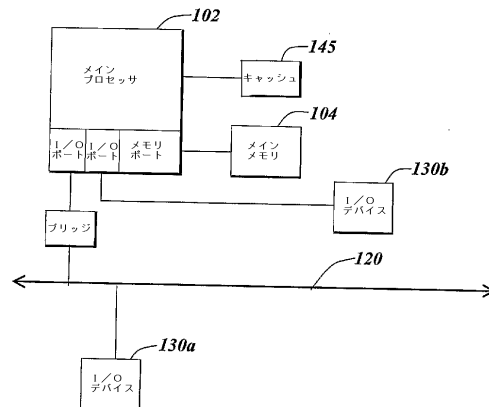
【図 1 B】

FIG. 1B



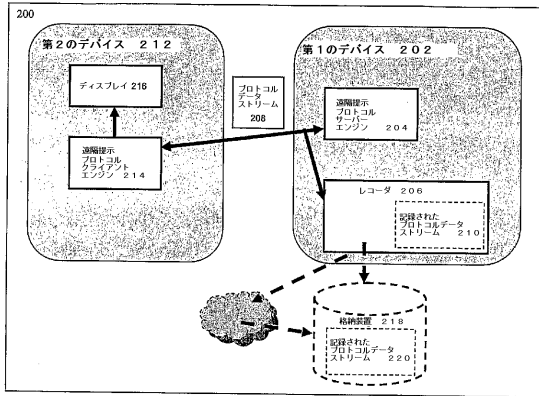
【図 1 C】

FIG. 1C



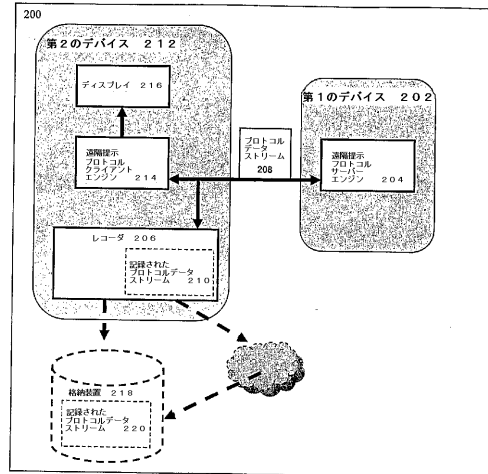
【図 2】

FIG. 2



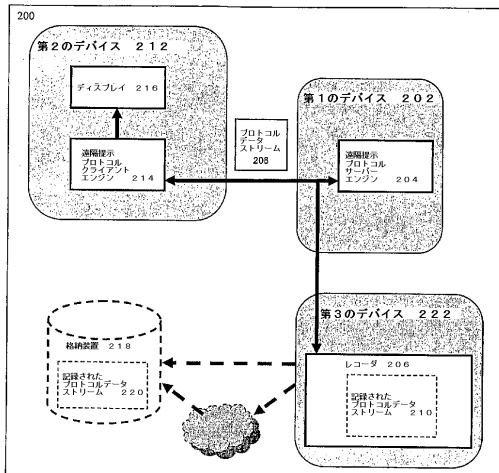
【図 3】

FIG. 3



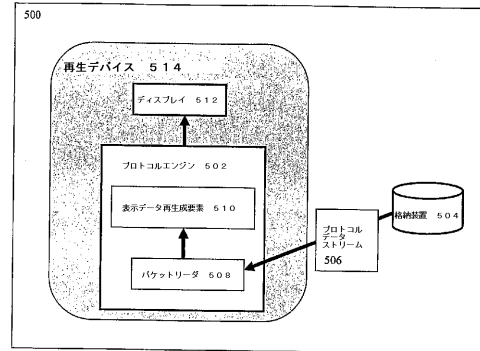
【図 4】

FIG. 4



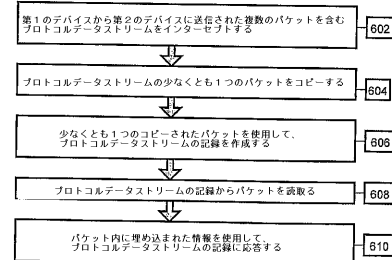
【図 5】

FIG. 5



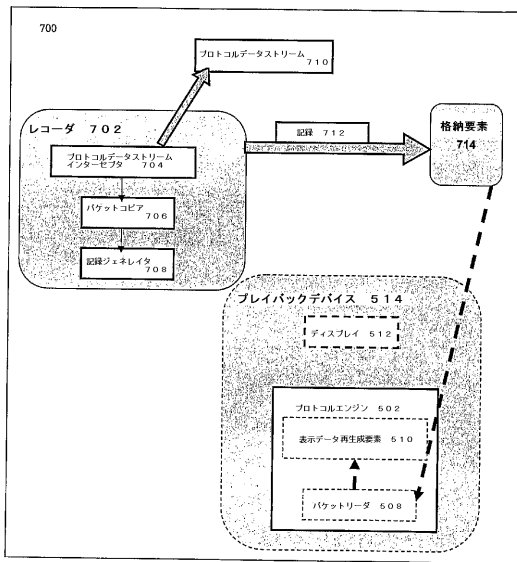
【図 6】

FIG. 6



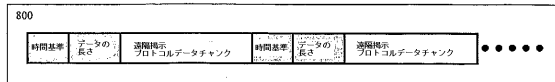
【図 7】

FIG. 7



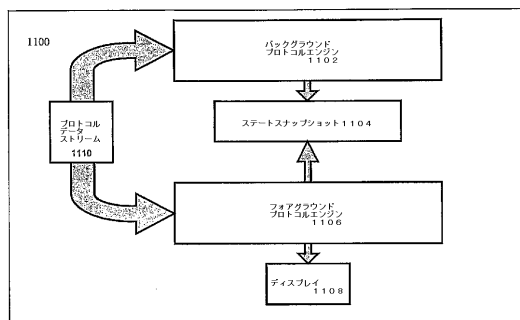
【図 8】

FIG. 8



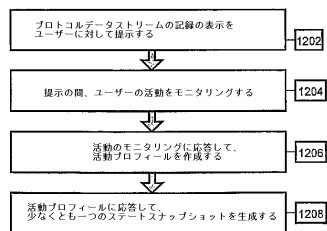
【図 11】

FIG. 11



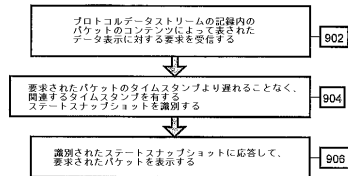
【図 12】

FIG. 12



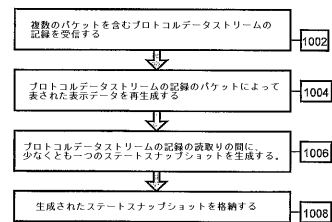
【図 9】

FIG. 9



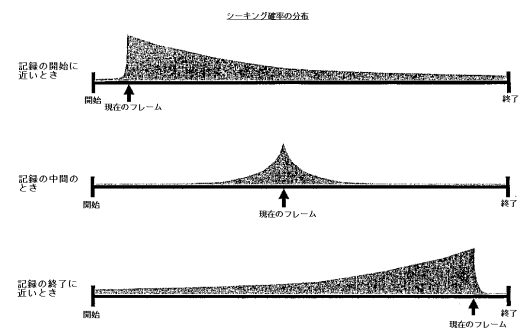
【図 10】

FIG. 10



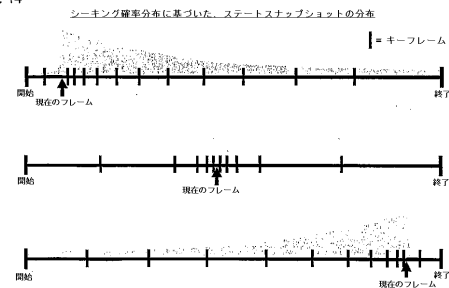
【図 13】

FIG. 13



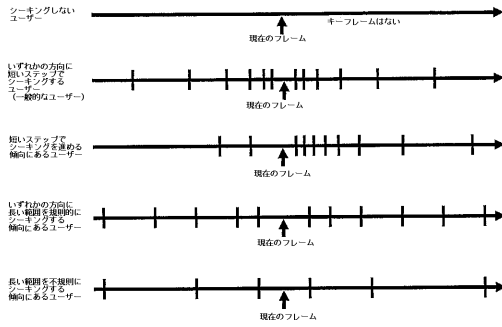
【図 14】

FIG. 14



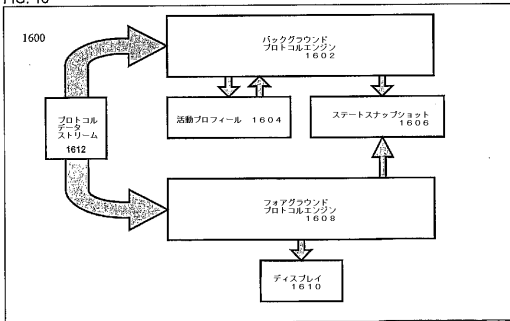
【図 15】

FIG. 15



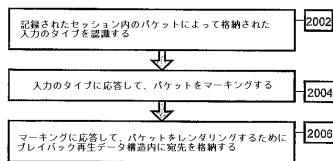
【図 16】

FIG. 16



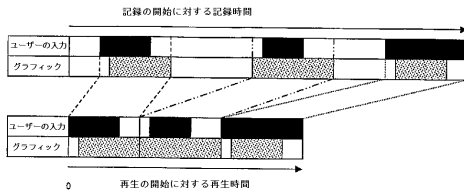
【図 20】

FIG. 20



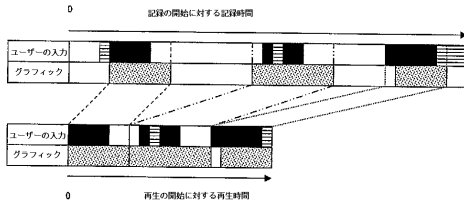
【図 21】

FIG. 21



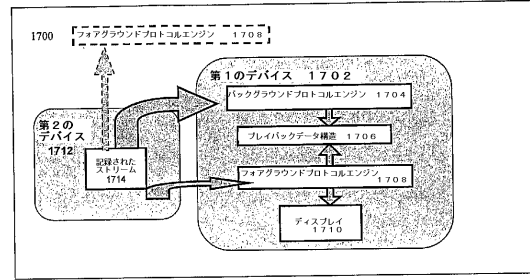
【図 22】

FIG. 22



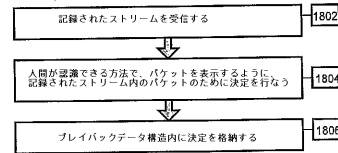
【図 17】

FIG. 17



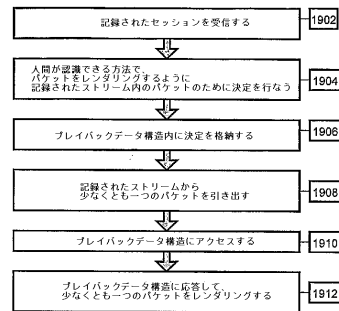
【図 18】

FIG. 18



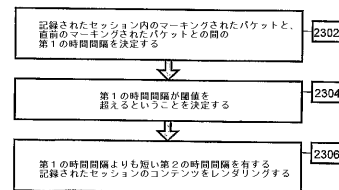
【図 19】

FIG. 19



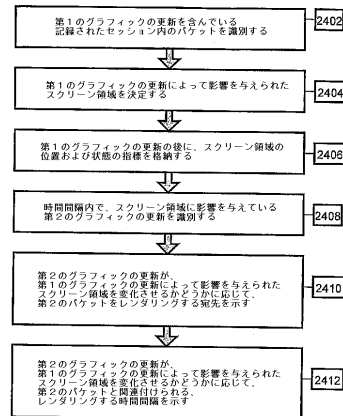
【図 23】

FIG. 23



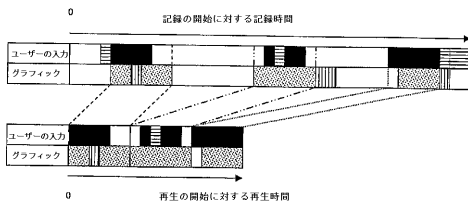
【図 24】

FIG. 24



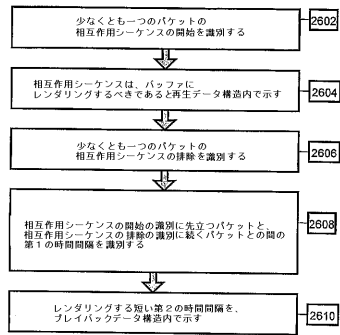
【図 25】

FIG. 25



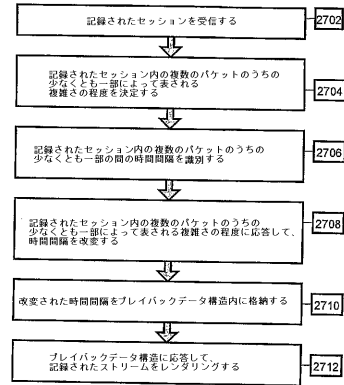
【図 26】

FIG. 26



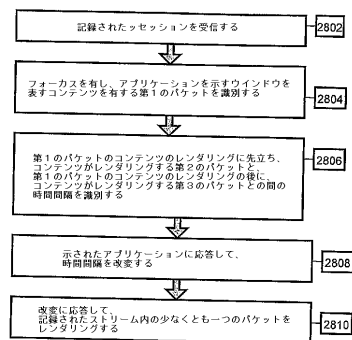
【図 27】

FIG. 27



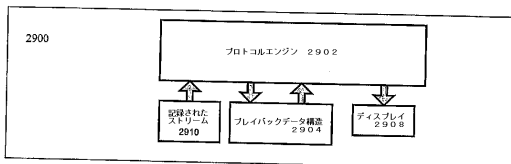
【図 28】

FIG. 28



【図 29】

FIG. 29



【 国際調査報告 】

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2006/000887

A. CLASSIFICATION OF SUBJECT MATTER INV. H04L12/26		
According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED Minimum documentation searched (classification system followed by classification symbols) H04L		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched		
Electronic data base consulted during the international search (name of data base and, where practical, search terms used) EPO-Internal, WPI Data, PAJ, IBM-TDB, INSPEC, COMPENDEX		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	SANDKLEF H: "Testing Applications with Xnee" LINUX JOURNAL, [Online] vol. 2004, no. 117, 2004, pages 1-6, XP002381423 ISSN: 1075-3583 ACM DIGITAL LIBRARY Retrieved from the Internet: URL: http://portal.acm.org/citation.cfm?id=959336.959341&coll=GUIDE&dl=ACM&idx=J508&part=periodical&wantType=periodical&title=Linux%20Journal [retrieved on 2006-05-17] the whole document ----- -/--	1-27
<input checked="" type="checkbox"/> Further documents are listed in the continuation of Box C.		
<input checked="" type="checkbox"/> See patent family annex.		
* Special categories of cited documents : "A" document defining the general state of the art which is not considered to be of particular relevance "E" earlier document but published on or after the international filing date "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) "O" document referring to an oral disclosure, use, exhibition or other means "P" document published prior to the international filing date but later than the priority date claimed "T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art. "Z" document member of the same patent family		
Date of the actual completion of the international search 4 July 2006		Date of mailing of the international search report 24/07/2006
Name and mailing address of the ISA/ European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016		Authorized officer Eraso Heiguera, J

INTERNATIONAL SEARCH REPORT

International application No

PCT/US2006/000887

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	STANONIK R: "Recording/Playing Both Gaze Date and Computer Interaction" WEB PAGE, [Online] 30 November 2000 (2000-11-30), XP002381424 Retrieved from the Internet: URL:http://hci.ucsd.edu/eye/gaze.txt> [retrieved on 2006-05-17] the whole document	1-27
A		28-33
X	SANDKLEF H: "Xnee Manual" MANUAL VERSION 1.08D, [Online] 3 October 2003 (2003-10-03), pages 1-36, XP002381190 INTERNET ARCHIVE Retrieved from the Internet: URL:http://web.archive.org/web/20040627125 613/www.gnu.org/software/xnee/www/manual/x nee.pdf> [retrieved on 2006-05-15] page 4 - page 6	1-27
A		41-50
X	STANONIK R: "Reversing the VNC Record File" INTERNET ARCHIVE, [Online] 18 March 2002 (2002-03-18), pages 1-3, XP002388536 Retrieved from the Internet: URL:http://web.archive.org/web/20060703115 503/http://hci.ucsd.edu/eye/reversing.txt> [retrieved on 2006-07-03] the whole document	68-73
A		34-40, 51-67
A	US 5 717 879 A (MORAN ET AL) 10 February 1998 (1998-02-10) abstract	1-73

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2006/000887

Box II Observations where certain claims were found unsearchable (Continuation of item 2 of first sheet)

This International Search Report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☐ Claims Nos.:
because they relate to subject matter not required to be searched by this Authority, namely:

2. ☐ Claims Nos.:
because they relate to parts of the International Application that do not comply with the prescribed requirements to such an extent that no meaningful International Search can be carried out, specifically:

3. ☐ Claims Nos.:
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box III Observations where unity of invention is lacking (Continuation of item 3 of first sheet)

This International Searching Authority found multiple inventions in this International application, as follows:

see additional sheet

1. ☒ As all required additional search fees were timely paid by the applicant, this International Search Report covers all searchable claims.

2. ☐ As all searchable claims could be searched without effort justifying an additional fee, this Authority did not invite payment of any additional fee.

3. ☐ As only some of the required additional search fees were timely paid by the applicant, this International Search Report covers only those claims for which fees were paid, specifically claims Nos.:

4. ☐ No required additional search fees were timely paid by the applicant. Consequently, this International Search Report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

Remark on Protest

- ☐ The additional search fees were accompanied by the applicant's protest.
- ☒ No protest accompanied the payment of additional search fees.

FURTHER INFORMATION CONTINUED FROM PCT/ISA/ 210

This International Searching Authority found multiple (groups of) inventions in this international application, as follows:

1. claims: 1-27

Methods and systems for generating playback instructions for playback of a recorded computer session.

2. claims: 28-33

Methods for eliminating a period of inactivity in rendering a recorded session.

3. claims: 34-40

A method of eliminating graphics updates in rendering a recorded session.

4. claims: 41-50

A method of eliminating interaction sequences in rendering a recorded session.

5. claims: 51-73

Methods and systems of automatic time-warped playback in rendering a recorded computer session.

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/US2006/000887

Patent document cited in search report		Publication date		Patent family member(s)	Publication date
US 5717879	A	10-02-1998	JP	9179712 A	11-07-1997

フロントページの続き

(81)指定国 AP(BW,GH,GM,KE,LS,MW,MZ,NA,SD,SL,SZ,TZ,UG,ZM,ZW),EA(AM,AZ,BY,KG,KZ,MD,RU,TJ,TM),EP(AT,BE,BG,CH,CY,CZ,DE,DK,EE,ES,FI,FR,GB,GR,HU,IE,IS,IT,LT,LU,LV,MC,NL,PL,PT,RO,SE,SI,SK,TR),OA(BF,BJ,CF,CG,CI,CM,GA,GN,GQ,GW,ML,MR,NE,SN,TD,TG),AE,AG,AL,AM,AT,AU,AZ,BA,BB,BG,BR,BW,BY,BZ,CA,CH,CN,CO,CR,CU,CZ,DE,DK,DM,DZ,EC,EE,EG,ES,FI,GB,GD,GE,GH,GM,HR,HU,ID,IL,IN,IS,JP,KE,KG,KM,KN,KP,KR,KZ,LC,LK,LR,LS,LT,LU,LV,LY,MA,MD,MG,MK,MN,MW,MX,MZ,NA,NG,NI,NO,NZ,OM,PG,PH,PL,PT,RO,RU,SC,SD,SE,SG,SK,SL,SM,SY,TJ,TM,TN,TR,TT,TZ,UA,UG,US,UZ,VC,VN,YU,ZA,ZM,ZW

(特許庁注：以下のものは登録商標)

- 1 . F R A M
- 2 . L i n u x

(72)発明者 マザフェリ , リチャード ジェームス

オーストラリア国 2 1 1 3 ノース ライド , ジュリアス アベニュー 1 , レベル 3 ,
サイトリックス システムズ オーストラレーシア アール アンド ディー ピーティーワイ
リミテッド 気付

Fターム(参考) 5K030 GA18 HB02 JT10 KA07 LD17 MC07