

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第6325001号
(P6325001)

(45) 発行日 平成30年5月16日 (2018. 5. 16)

(24) 登録日 平成30年4月20日 (2018. 4. 20)

(51) Int. Cl.

F I

G 0 6 F 9/54 (2006. 01)

G 0 6 F 9/46 4 8 0 C

G 0 6 F 9/48 (2006. 01)

G 0 6 F 9/46 4 5 7

請求項の数 18 (全 20 頁)

(21) 出願番号 特願2015-555199 (P2015-555199)
 (86) (22) 出願日 平成26年1月17日 (2014. 1. 17)
 (65) 公表番号 特表2016-508638 (P2016-508638A)
 (43) 公表日 平成28年3月22日 (2016. 3. 22)
 (86) 国際出願番号 PCT/US2014/012121
 (87) 国際公開番号 W02014/116527
 (87) 国際公開日 平成26年7月31日 (2014. 7. 31)
 審査請求日 平成29年1月16日 (2017. 1. 16)
 (31) 優先権主張番号 13/748, 173
 (32) 優先日 平成25年1月23日 (2013. 1. 23)
 (33) 優先権主張国 米国 (US)

早期審査対象出願

(73) 特許権者 508178054
 フェイスブック, インク.
 アメリカ合衆国 カリフォルニア 940
 25, メンロー パーク, ウィロー ロード 1601
 (74) 代理人 100105957
 弁理士 恩田 誠
 (74) 代理人 100068755
 弁理士 恩田 博宣
 (74) 代理人 100142907
 弁理士 本田 淳
 (72) 発明者 リュー、フィリップ
 アメリカ合衆国 94025 カリフォル
 ニア州 メンロー パーク ウィロー ロ
 ード 1601

最終頁に続く

(54) 【発明の名称】 階層データ構造のノードにおいて再帰的イベントリスナを用いる方法およびシステム

(57) 【特許請求の範囲】

【請求項 1】

コンピュータネットワークにおけるコンピュータシステムにおいて、前記コンピュータシステムのクライアントから、起点ノードにイベントリスナを登録するための要求を受信することであって、前記起点ノードが、階層中で互いに関連する複数のノードのうちの1つであり、前記複数のノードが、それぞれ前記コンピュータシステムの記憶装置の論理区画を表し、前記記憶装置が、複数の論理区画を含み、前記複数の論理区画がそれぞれ、前記記憶装置に含まれたデータの別個の非重複部分を含む、前記要求を受信すること、

前記コンピュータシステムのサーバによって、前記起点ノードに前記イベントリスナを登録することであって、前記イベントリスナが、前記起点ノードにおける第1のイベントの発生を前記クライアントに通知するように構成されている、前記イベントリスナを登録すること、

前記サーバによって、前記起点ノードが前記階層中に子孫ノードを有するか否かを判定すること、

前記起点ノードが前記階層中に子孫ノードを有する旨の判定に回答して、前記サーバにより、前記子孫ノードに別のイベントリスナを登録する必要なく、前記子孫ノードにおける第2のイベントの発生を前記クライアントに通知するように前記イベントリスナを設定することを備える、方法。

【請求項 2】

前記起点ノードにおける前記第1のイベントまたは前記子孫ノードにおける前記第2の

10

20

イベントの少なくとも一方の発生時に、前記第 1 のイベントまたは前記第 2 のイベントの少なくとも一方の発生を前記クライアントに通知することをさらに備える、請求項 1 に記載の方法。

【請求項 3】

前記起点ノードに前記イベントリスナを登録することは、

前記第 1 のイベントまたは前記第 2 のイベントの発生を前記クライアントに通知した後も前記イベントリスナを保持すること、

前記クライアントが削除するまで前記イベントリスナを保持することを含む、請求項 2 に記載の方法。

【請求項 4】

前記第 1 のイベントまたは前記第 2 のイベントの発生を前記クライアントに通知することは、

前記第 1 のイベントの発生を受信することであって、前記サーバにより、前記起点ノードに対応する前記サーバのメモリにおけるフィールドが、前記起点ノードへの前記イベントリスナの登録を示す所定値に設定されたか否かを判定する、前記第 1 のイベントの発生を受信すること、

前記起点ノードに対応する前記サーバのメモリにおけるフィールドが、前記子孫ノードへの前記イベントリスナの登録を示す所定値に設定されたか否かを判定する必要なく、前記第 2 のイベントの発生を受信すること、

前記起点ノードにおける前記第 1 のイベントの発生または前記子孫ノードにおける前記第 2 のイベントの発生を前記クライアントに通知することを含む、請求項 2 に記載の方法。

【請求項 5】

前記起点ノードに前記イベントリスナを登録することは、

所定の種類の前記第 1 のイベントまたは前記第 2 のイベントの発生を前記クライアントに通知するように前記イベントリスナを構成することを含む、請求項 1 に記載の方法。

【請求項 6】

前記所定の種類の前記第 1 のイベントまたは前記第 2 のイベントが、(i) 新たなノードの作成、(i i) 既存ノードの消去、または(i i i) 前記既存ノードの修正のうちの少なくとも 1 つを含む、請求項 5 に記載の方法。

【請求項 7】

前記第 1 のイベントおよび前記第 2 のイベントが、同じ所定の種類のイベントである、請求項 1 に記載の方法。

【請求項 8】

前記第 1 のイベントおよび前記第 2 のイベントが、異なる所定の種類のイベントである、請求項 1 に記載の方法。

【請求項 9】

前記子孫ノードにおける前記第 2 のイベントの発生を前記クライアントに通知するように前記イベントリスナを設定することは、

前記階層中の前記起点ノードから始まる選択経路に沿った子孫ノードにおけるイベントの発生を前記クライアントに通知するように前記イベントリスナを構成することを含む、請求項 1 に記載の方法。

【請求項 10】

前記起点ノードに前記イベントリスナを登録することは、

アクセス制御を設定することによって、前記起点ノードへの前記イベントリスナの登録を前記コンピュータシステムにおける予め設定された組のユーザ役割に制限することを含む、請求項 1 に記載の方法。

【請求項 11】

方法であって、

コンピュータネットワークにおけるコンピュータシステムによって、前記コンピュータ

10

20

30

40

50

システムのサーバ中の起点ノードにおいてイベントが発生したか否かを判定することであって、前記起点ノードが、階層中で互いに関連する複数のノードのうちの1つであり、前記複数のノードが、それぞれ前記コンピュータシステム中の記憶装置の論理区画を表し、前記記憶装置が、複数の論理区画を含み、前記複数の論理区画がそれぞれ、前記記憶装置に含まれたデータの別個の非重複部分を含む、前記イベントが発生したか否かを判定すること、

前記起点ノードにおいて前記イベントが発生した旨の判定に応答して、前記サーバにより、前記コンピュータシステムのクライアントによって前記起点ノードにイベントリスナが登録されたか否かを判定すること、

前記起点ノードに前記イベントリスナが登録された旨の判定に応答して、前記起点ノードにおける前記イベントの発生を前記クライアントに通知すること、

前記起点ノードに前記イベントリスナが登録されていない旨の判定に応答して、前記サーバにより、前記起点ノードが前記階層中に先祖ノードを有するか否かを判定すること、

前記起点ノードが前記階層中に先祖ノードを有する旨の判定に応答して、前記サーバにより、前記先祖ノードに前記イベントリスナが登録されたか否かを判定すること、

前記先祖ノードに前記イベントリスナが登録された旨の判定に応答して、前記起点ノードにおける前記イベントの発生を前記クライアントに通知することを備える、方法。

【請求項12】

前記起点ノードに登録された前記イベントリスナが、前記イベントの発生を前記クライアントに通知する動作の後でも保持される、請求項11に記載の方法。

【請求項13】

前記イベントが、(i)新たなノードの作成、(ii)既存ノードの消去、または(iii)前記既存ノードの修正のうち少なくとも1つを含む、請求項11に記載の方法。

【請求項14】

前記サーバと前記クライアントとの間の通信障害に응答して、前記クライアントから、新たなサーバ中の前記起点ノードに新たなイベントリスナを登録するための要求を受信することであって、前記新たなサーバが、前記階層中の前記複数のノードの少なくとも一部のコピーを有し、前記要求が、前記サーバから受信した最新のイベントの処理番号を含む、前記要求を受信すること、

前記新たなサーバにより、前記処理番号を用いることによって、前記最新のイベントを受信した時刻から前記起点ノードおよび前記起点ノードの子孫ノードで発生したイベントの一覧を取得すること、

前記新たなサーバによって、前記イベントの一覧を前記クライアントに送信することをさらに備える、請求項11に記載の方法。

【請求項15】

装置であって、

サーバと、

前記サーバと協働することにより、クライアントから、起点ノードにイベントリスナを登録するための要求を受信するように構成されている受信モジュールであって、前記起点ノードが、階層中で互いに接続される複数のノードのうちの1つであり、前記複数のノードが、それぞれ記憶ユニットの論理区画を表し、前記記憶ユニットが、複数の論理区画を含み、前記複数の論理区画がそれぞれ、前記記憶ユニットに含まれたデータの別個の非重複部分を含む、前記受信モジュールと、

前記受信モジュールと協働することにより、前記起点ノードに前記イベントリスナを登録するように構成されている登録モジュールであって、前記イベントリスナが、前記起点ノードにおける第1のイベントの発生を前記クライアントに通知するように構成されている、前記登録モジュールと、

前記登録モジュールと協働することにより、前記起点ノードが前記階層中に子孫ノードを有するか否かを判定するように構成されている階層判定モジュールと、

前記階層判定モジュールと協働することにより、前記起点ノードが前記階層中に子孫ノ

10

20

30

40

50

ードを有する旨の判定に応答して、前記子孫ノードに別のイベントリスナを登録する必要なく、前記子孫ノードにおける第2のイベントの発生を前記クライアントに通知するように前記イベントリスナを設定するように構成されている階層イベント処理モジュールとを備える、装置。

【請求項16】

前記イベントリスナは、前記登録モジュールと協働することにより、前記起点ノードにおける前記第1のイベントまたは前記子孫ノードにおける前記第2のイベントの少なくとも一方の発生時に、前記第1のイベントまたは前記第2のイベントの少なくとも一方の発生を前記クライアントに通知するようにさらに構成されている、請求項15に記載の装置。

10

【請求項17】

前記イベントリスナは、

前記第1のイベントの発生を受信することであって、前記サーバにより、前記起点ノードに対応する前記サーバのメモリ中のビットが、前記起点ノードへの前記イベントリスナの登録を示す所定値に設定されたか否かを判定することを含む、前記第1のイベントの発生を受信すること、

前記起点ノードに対応する前記サーバのメモリ中のビットが、前記子孫ノードへの前記イベントリスナの登録を示す所定値に設定されたか否かを判定する必要なく、前記第2のイベントの発生を受信すること、

前記起点ノードにおける前記第1のイベントの発生または前記子孫ノードにおける前記第2のイベントの発生を前記クライアントに通知することを実施するようにさらに構成されている、請求項16に記載の装置。

20

【請求項18】

前記イベントリスナは、所定の種類のイベントの発生を通知するようにさらに構成され、

前記所定の種類の前記イベントは、(i)新たなノードの作成、(ii)既存ノードの消去、または(iii)前記既存ノードの修正のうち少なくとも1つを含む、請求項15に記載の装置。

【発明の詳細な説明】

【技術分野】

30

【0001】

本発明は一般的に、コンピュータアプリケーションにおけるイベントリスナに関する。より詳細には、本発明は、階層データ構造のノードにおける再帰的イベントリスナの使用に関する。

【背景技術】

【0002】

コンピュータ関連技術において、イベントリスナは、ソフトウェアオブジェクト(以下、「オブジェクト」と称する)における特定の1つまたは複数の種類のイベントの発生を検出し(「受信し」)、このイベントの発生を加入者に通知する要素である。異なる階層レベルにオブジェクトを有する階層データ構造において、これらオブジェクトのいずれかにおけるイベントの発生(オブジェクトの作成、オブジェクトの消去、またはオブジェクトの修正等)を検出するため、現行のイベントリスナ技術では、階層データ構造のオブジェクトそれぞれにイベントリスナを登録する。これらの技術は通常、イベントリスナの登録にかなりの時間およびメモリを消費する。

40

【0003】

たとえば、ソーシャルネットワーキング環境が100,000個のユーザ(クライアント)と、何らかのユーザ情報を表す階層データ構造に構成された1億個のオブジェクトを含むサーバとを有するシナリオを考える。また、オブジェクトへのイベントリスナの登録に1ビットのメモリを消費するものと仮定する。100,000個のクライアントが存在し、これら100,000個のクライアントがそれぞれ、サーバ上の1億個のオブジェク

50

トそれぞれにイベントリスナを配置することを望む場合は、イベントリスナの登録だけに1.25テラバイトのメモリが消費される可能性がある。オブジェクトには、(a)オブジェクトの作成、(b)オブジェクトの修正、および(c)オブジェクトの子という3種類のイベントリスナを登録するようにしてもよい。このため、イベントリスナの登録だけに3.75テラバイトのメモリが消費される可能性がある。さらに、階層中のオブジェクトそれぞれに対する別個の登録要求をクライアントが送信する場合は、登録にかなりのネットワーク帯域幅および時間が消費される可能性がある。現行のイベントリスナ技術は通常、少なくとも時間、メモリ、およびネットワーク帯域幅の消費に関しては非効率的である。さらに、現行のイベントリスナ登録技術に関する非効率性は、クライアント数、階層中のオブジェクト数、またはその両者の増加に応じて、大幅に拡大する可能性がある。分散システムアーキテクチャにおいては、新たに分散されるアプリケーションまたはサービスと、これらのサービスの消費を望む新たなクライアントとが絶え間なく増加する可能性がある。このような分散環境において、現行のイベントリスナ登録技術では通常、イベントリスナの登録にかなりのリソースを消費することでシステム全体の効率が大幅に低下することになる。

10

【発明の概要】

【課題を解決するための手段】

【0004】

以下、階層データ構造のノードに再帰的イベントリスナを登録する技術を説明する。再帰的イベントリスナは、当該再帰的イベントリスナが登録されたノードで発生するイベントを受信するとともに、当該再帰的イベントリスナまたはその他任意のイベントリスナを当該ノードの子孫ノードに登録する必要なく、当該子孫ノードで発生するイベントも受信するイベントリスナである。この技術は、特に階層データ構造を有する環境において、メモリおよび処理リソースをより効率的に使用するため、監視対象のオブジェクトの数または監視を要求するクライアントの数による大きな悪影響を受けることはない。

20

【0005】

開示技術の一実施形態は、クライアントから、起点ノードにイベントリスナを登録する要求を受信するステップを含む。起点ノードは、階層中で互いに関連する複数のノードのうちの1つであり、当該ノードがそれぞれ、記憶装置の論理区画を表す。サーバは、起点ノードにイベントリスナを登録し、当該イベントリスナは、起点ノードにおける第1のイベントの発生をクライアントに通知するように構成されている。さらに、サーバは、起点ノードが階層中に子孫ノードを有するか否かを判定し、起点ノードが階層中に子孫ノードを有する旨の判定に回答して、子孫ノードに別のイベントリスナを登録する必要なく、子孫ノードにおける第2のイベントの発生をクライアントに通知するようにイベントリスナを設定する。

30

【0006】

開示技術の少なくとも一部の実施形態において、クライアントは、ノードにおける第1のイベントまたは子孫ノードにおける第2のイベントの少なくとも一方の発生時に、第1のイベントまたは第2のイベントの少なくとも一方の発生を通知される。

【0007】

開示技術の少なくとも一部の実施形態において、ノードに登録されたイベントリスナは、第1のイベントまたは第2のイベントの発生をクライアントに通知した後も保持される。

40

【0008】

本発明の一部の実施形態は、上記の追加または代替として、他の態様、要素、特徴、およびステップを有する。これらの潜在的な追加および代替については、本明細書のその他の部分で説明する。

【0009】

本発明に係る実施形態は、方法、記憶媒体、およびシステムに関する添付の特許請求の範囲において特に開示しており、方法等の特許請求の範囲のあるカテゴリに記載された任

50

意の特徴は、システム等の特許請求の範囲の別のカテゴリでも同様に請求可能である。

【0010】

本発明に係る一実施形態において、方法は、

コンピュータネットワーク中のコンピュータシステムにおいて、当該コンピュータシステム中のクライアントから、起点ノードにイベントリスナを登録する要求を受信するステップであって、起点ノードが階層中で互いに関連する複数のノードのうちの1つであり、当該ノードがそれぞれコンピュータシステム中の記憶装置の論理区画を表す、前記受信するステップと、

コンピュータシステム中のサーバによって、起点ノードにイベントリスナを登録するステップであって、イベントリスナが、起点ノードにおける第1のイベントの発生をクライアントに通知するように構成された、前記登録するステップと、

サーバによって、起点ノードが階層中に子孫ノードを有するか否かを判定するステップと、

起点ノードが階層中に子孫ノードを有する旨の判定に応答して、サーバにより、子孫ノードに別のイベントリスナを登録する必要なく、子孫ノードにおける第2のイベントの発生をクライアントに通知するようにイベントリスナを設定するステップと、

を含む。

【0011】

別の実施形態において、この方法は、起点ノードにおける第1のイベントまたは子孫ノードにおける第2のイベントの少なくとも一方の発生時に、第1のイベントまたは第2のイベントの少なくとも一方の発生をクライアントに通知するステップをさらに含む。

【0012】

起点ノードにイベントリスナを登録するステップは、

第1のイベントまたは第2のイベントの発生をクライアントに通知した後もイベントリスナを保持するサブステップと、

クライアントが削除するまでイベントリスナを保持するサブステップと、
を含むことができる。

【0013】

第1のイベントまたは第2のイベントの発生をクライアントに通知するステップは、

第1のイベントの発生を受信するサブステップであって、サーバにより、起点ノードに対応するサーバのメモリ中のフィールドが、起点ノードへのイベントリスナの登録を示す所定値に設定されたか否かを判定するサブサブステップを含む、前記受信するサブステップと、

起点ノードに対応するサーバのメモリ中のフィールドが、子孫ノードへのイベントリスナの登録を示す所定値に設定されたか否かを判定する必要なく、第2のイベントの発生を受信するサブステップと、

起点ノードにおける第1のイベントの発生または子孫ノードにおける第2のイベントの発生をクライアントに通知するサブステップと、

を含むことができる。

【0014】

起点ノードにイベントリスナを登録するステップは、所定の種類の第1のイベントまたは第2のイベントの発生をクライアントに通知するようにイベントリスナを構成するサブステップを含むことができる。

【0015】

所定の種類の第1のイベントまたは第2のイベントは、(i)新たなノードの作成、(ii)既存ノードの消去、または(iii)既存ノードの修正のうちの少なくとも1つを含むことができる。

【0016】

第1のイベントおよび第2のイベントは、同じ所定の種類のイベントとすることができる。

第1のイベントおよび第2のイベントは、異なる所定の種類のイベントとすることができる。

【0017】

記憶装置は、複数の論理区画を含ことができ、当該論理区画がそれぞれ、記憶装置に含まれた別個の非重複データの一部を含む。

子孫ノードにおける第2のイベントの発生をクライアントに通知するようにイベントリスナを設定するステップは、階層中の起点ノードから始まる選択経路に沿った子孫ノードにおけるイベントの発生をクライアントに通知するようにイベントリスナを構成するサブステップを含むことができる。

【0018】

起点ノードにイベントリスナを登録するステップは、アクセス制御を設定することによって、起点ノードへのイベントリスナの登録をコンピュータシステム中の所定のユーザ役割セットに制限するサブステップを含むことができる。

【0019】

本発明に係るさらに別の実施形態において、方法は、

コンピュータネットワーク中のコンピュータシステムによって、コンピュータシステムのサーバ中の起点ノードにおいてイベントが発生したか否かを判定するステップであって、起点ノードが階層中で互いに関連する複数のノードのうちの1つであり、当該ノードがそれぞれコンピュータシステム中の記憶装置の論理区画を表す、前記判定するステップと、

起点ノードにおいてイベントが発生した旨の判定に回答して、サーバにより、コンピュータシステム中のクライアントによって起点ノードにイベントリスナが登録されたか否かを判定するステップと、

起点ノードにイベントリスナが登録された旨の判定に回答して、起点ノードにおけるイベントの発生をクライアントに通知するステップと、

起点ノードにイベントリスナが登録されていない旨の判定に回答して、サーバにより、起点ノードが階層中に先祖ノードを有するか否かを判定するステップと、

起点ノードが階層中に先祖ノードを有する旨の判定に回答して、サーバにより、先祖ノードにイベントリスナが登録されたか否かを判定するステップと、

先祖ノードにイベントリスナが登録された旨の判定に回答して、起点ノードにおけるイベントの発生をクライアントに通知するステップと、

を含む。

【0020】

起点ノードに登録されたイベントリスナは、イベントの発生をクライアントに通知する動作の後でも保持可能である。

イベントは、(i)新たなノードの作成、(ii)既存ノードの消去、または(iii)当該既存ノードの修正のうちの少なくとも1つを含むことができる。

【0021】

さらに別の実施形態において、この方法は、

サーバとクライアントとの間の通信障害に回答して、クライアントから、新たなサーバ中の起点ノードに新たなイベントリスナを登録する要求を受信するステップであって、新たなサーバが階層中のノードの少なくとも一部のコピーを有し、上記要求がサーバから受信した最新のイベントの処理番号を含む、前記受信するステップと、

新たなサーバにより、処理番号を用いることによって、最新のイベントを受信した時刻から起点ノードおよび当該起点ノードの子孫ノードで発生したイベントの一覧を取得するステップと、

新たなサーバによって、イベントの一覧をクライアントに送信するステップと、を含む。

【0022】

同様に請求可能な本発明の一実施形態において、装置は、

10

20

30

40

50

サーバと、

サーバ装置と協働することにより、クライアントから、起点ノードにイベントリスナを登録する要求を受信するように構成された受信モジュールであって、起点ノードが階層中で互いに接続される複数のノードのうちの1つであり、当該ノードがそれぞれ記憶ユニットの論理区画を表す、受信モジュールと、

受信モジュールと協働することにより、起点ノードにイベントリスナを登録するように構成された登録モジュールであって、イベントリスナが、起点ノードにおける第1のイベントの発生をクライアントに通知するように構成された、登録モジュールと、

登録モジュールと協働することにより、起点ノードが階層中に子孫ノードを有するか否かを判定するように構成された階層判定モジュールと、

階層判定モジュールと協働することにより、起点ノードが階層中に子孫ノードを有する旨の判定に回答して、子孫ノードに別のイベントリスナを登録する必要なく、子孫ノードにおける第2のイベントの発生をクライアントに通知するようにイベントリスナを設定するように構成された階層イベント処理モジュールと、

を備える。

【0023】

イベントリスナは、登録モジュールと協働することにより、起点ノードにおける第1のイベントまたは子孫ノードにおける第2のイベントの少なくとも一方の発生時に、第1のイベントまたは第2のイベントの少なくとも一方の発生をクライアントに通知するようにさらに構成可能である。

【0024】

イベントリスナは、

第1のイベントの発生を受信し、当該第1のイベントの発生を受信することが、サーバにより、起点ノードに対応するサーバのメモリ中のビットが、起点ノードへのイベントリスナの登録を示す所定値に設定されたか否かを判定することを含み、

起点ノードに対応するサーバのメモリ中のビットが、子孫ノードへのイベントリスナの登録を示す所定値に設定されたか否かを判定する必要なく、第2のイベントの発生を受信し、

起点ノードにおける第1のイベントの発生または子孫ノードにおける第2のイベントの発生をクライアントに通知する、

ようにさらに構成可能である。

【0025】

イベントリスナは、所定の種類のイベントの発生を通知するようにさらに構成可能であり、当該所定の種類のイベントが、(i)新たなノードの作成、(ii)既存ノードの消去、または(iii)既存ノードの修正のうちの少なくとも1つを含む。

【0026】

記憶ユニットは、複数の論理区画を含むことができ、当該論理区画がそれぞれ、記憶ユニットに含まれた別個の非重複データの一部を含む。

本発明のさらに別の実施形態において、1つまたは複数のコンピュータ可読持続性記憶媒体は、実行時に本発明または上記実施形態のうちのいずれかに係る方法を実施するように動作可能なソフトウェアを具現化する。

【0027】

本発明のさらに別の実施形態において、システムは、1つまたは複数のプロセッサと、当該プロセッサに結合され、当該プロセッサにより実行可能な命令を含むメモリとを備え、プロセッサは、命令の実行時に本発明または上記実施形態のうちのいずれかに係る方法を実施するように動作可能である。

【図面の簡単な説明】

【0028】

【図1】本発明の一実施形態が動作可能な環境を示した図である。

【図2】階層データ構造のノードにおける再帰的イベントリスナを示したブロック図であ

10

20

30

40

50

る。

【図3】階層データ構造のノードに再帰的イベントリスナを登録するためのフローチャートである。

【図4】階層データ構造のノードにおけるイベントの発生を通知するためのフローチャートである。

【図5】図1のコーディネーションサービスにおける再帰的イベントリスナの登録の一例を示した図である。

【図6】クライアントとサーバとの間の通信障害に回答して、クライアントにより再帰的イベントリスナを再登録するプロセスを示したフローチャートである。

【図7】様々な動作を実行するとともに、このような動作で生成および使用される、あるいはこのような動作で生成または使用される様々な情報を格納可能な装置のブロック図である。

【発明を実施するための形態】

【0029】

本説明における「一実施形態」等の言及は、説明している特定の特徴、機能、または特性が本発明の少なくとも1つの実施形態に含まれることを意味する。本明細書におけるこのような表現の使用は、必ずしもすべてが同じ実施形態を参照しているわけでもなければ、必ずしも相互に排他的でもない。

【0030】

階層データ構造のノードに再帰的イベントリスナを登録する方法およびシステムを開示する。開示技術の一実施形態は、クライアントから、起点ノードにイベントリスナを登録する要求を受信するステップを含む。起点ノードは、階層中で互いに関連する複数のノードのうちの1つであり、当該ノードがそれぞれ、たとえば記憶装置の論理区画を表す。サーバは、起点ノードにイベントリスナを登録し、当該イベントリスナは、起点ノードにおける第1のイベントの発生をクライアントに通知するように構成されている。さらに、サーバは、起点ノードが階層中に子孫ノード(descendant node)を有するか否かを判定し、起点ノードが階層中に子孫ノードを有する旨の判定に回答して、子孫ノードに別のイベントリスナを登録する必要なく、子孫ノードにおける第2のイベントの発生をクライアントに通知するようにイベントリスナを設定する。

【0031】

以下、特に断りのない限り、用語「イベントリスナ(event listener)」は、「再帰的イベントリスナ(recursive event listener)」も意味する。非再帰的イベントリスナについては、「非再帰的イベントリスナ(non-recursive event listener)」と称する。

【0032】

環境

図1は、開示技術の一実施形態が動作可能な環境100を示している。図示の通り、環境100は、たとえばメリーランド州フォレストヒルのアパッチソフトウェア財団(Apache Software Foundation)から入手可能なZooKeeper(商標)サービスの実施等が考えられるコーディネーションサービス125を含む。別の実施形態において、環境100は、分散システムまたは分散アプリケーションに対してコーディネーションサービスを提供するその他類似のサービスを含んでいてもよい。コーディネーションサービス(coordination service)125は、分散アプリケーションに対して、たとえば同期、構成保守、グループ、および命名等のコーディネーションサービスを提供する。コーディネーションサービス125は、サーバグループ105にわたって複製され、アンサンブル(ensemble)と称する(以下、サーバ120のグループ105は、「グループ」または「アンサンブル」と称する場合もある)。グループ105のサーバの大部分が利用可能である限り、コーディネーションサービス125も利用可能である。グループ105の各サーバは、たとえばその他各サーバの状態等の情報を有していてもよい。

【0033】

アンサンブル 105 において、あるサーバは、たとえば合意による書き込み等の処理の承認および調整を役割とするリーダー (leader) 115 として動作してもよい。アンサンブルにおけるリーダー 115 以外のサーバはフォロワ (follower) として動作してもよく、フォロワは、リーダー 115 の直接的な読み出し専用の複製であってもよい。コーディネーションサービス 125 において、クライアントからの一部の書き込み要求は、リーダー 115 に転送される。フォロワは、リーダー 115 からの提案を受信するとともに、クライアントからの要求に対するサービス提供に賛成または反対してもよい。さらに、コーディネーションサービス 125 において、リーダー 115 が故障あるいは無効となった場合は、その他任意のフォロワが合意により新たなリーダーとなってもよく、また、要求に対するサービス提供を継続してもよい。その他のフォロワは、この新たなリーダーと同期していてもよい。

10

【0034】

環境 100 は、コーディネーションサービス 125 を用いるアプリケーション (図示せず) が使用するデータを含むデータベース (図示せず) 等の記憶媒体を具備する。たとえばソーシャルネットワーキング環境等のアプリケーションにおいて、データベースは、たとえばユーザプロファイルデータを含んでいてもよい。また、データベースは、「シャード (shard)」とも称する多くの論理区画に分割され、それぞれがデータベース中のデータの一部分を含んでいてもよい。たとえば、ソーシャルネットワーキングアプリケーションにおいて、データベースが 100,000 個のユーザのユーザプロファイルデータを含む場合は、第 1 のシャードがユーザ 1 ~ 10,000 のユーザプロファイルデータを含み、第 2 のシャードがユーザ 10,001 ~ 20,000 のユーザプロファイルデータを含んでいてもよく、以下同様である。データベースのシャードはそれぞれ、データベースの別個の非重複データの一部分を含む。

20

【0035】

データベースのシャードは、ツリー構造 130 等の階層データ構造のノードセットとして表すことも可能である。ツリー構造 130 は、たとえばサーバ 120 のメモリ中に作成される。クライアント 110 は、サーバ 120 上のツリー構造 130 の対応するノードにアクセスすることによって、データベースのシャードのデータにアクセスする。ツリー構造 130 のルートノード「A」は、ユーザデータベースを表していてもよく、ノード「B」は、ユーザ 1 ~ 10,000 のユーザデータを有するシャードを表していてもよく、ノード「C」は、ユーザ 10,001 ~ 20,000 のユーザデータを有するシャードを表していてもよく、ノード「D」は、ユーザ 20,001 ~ 30,000 のユーザデータを有するシャードを表していてもよい。

30

【0036】

クライアント 110 は、ツリー構造 130 のノードのいずれかにおけるイベントの発生の通知を受けてもよい。クライアント 110 は、ノードにおけるイベントの発生の通知を受けるため、特定ノードにイベントリスナを登録するようサーバ 120 に要求可能である。たとえば、クライアント 110 は、ツリー構造 130 のノード A にイベントリスナを登録するようサーバ 120 に要求する。イベントリスナは、再帰的イベントリスナであってもよいし、非再帰的イベントリスナであってもよい。

【0037】

40

非再帰的イベントリスナは、当該イベントリスナが登録されたノードにおいてのみ発生しているイベントを監視する。たとえば、クライアント 110 は、ノード A に非再帰的イベントリスナを登録している場合、ノード A においてのみイベントが発生している場合に、当該イベントの発生の通知を受けることになる。クライアント 110 は、ノード A の子孫ノードで発生しているイベントの通知は受けない (一部の実施形態において、クライアントは、ノードの子が変化している場合に通知を受けるようにしてもよいが、たとえば最初の 1 回のみ通知を受けるようにしてもよい)。一方、クライアント 110 は、ノード A に再帰的イベントリスナを登録している場合、ツリー構造 130 のノード A または当該ノード A の子孫ノードにおいて発生しているイベントについて、その発生の通知を受けることになる。たとえば、クライアント 110 は、ノード A に再帰的イベントリスナを登録し

50

ている場合、ノードAおよび当該ノードAの子孫ノードすなわちノードB～G、あるいはノードAまたはノードB～Gのいずれかにおいて発生しているイベントについて、その発生の通知を受けることになる。同様に、ノードBに再帰的イベントリスナが登録されている場合、クライアント110は、ノードBまたはその子孫ノードEおよびFのいずれかにおいて発生しているイベントについて、その発生の通知を受けることになる。

【0038】

ノードに再帰的イベントリスナを登録すると、当該ノードまたはその子孫ノードにおけるイベントの発生が加入者に通知されるため、この技術では、クライアントが当該ノードの子孫ノードにイベントリスナを登録する必要がない。これにより、すべての子孫ノードに対する非再帰的イベントリスナの登録に消費される時間およびネットワーク帯域幅が抑えられる。さらに、再帰的イベントリスナの登録の格納に消費されるメモリについても、抑えられる場合がある。単一の登録を格納することによって、ノードおよびその子孫ノードの監視が可能となるためである。一方、非再帰的な登録においては通常、クライアント110が監視したいノードと同数の登録が行われる。したがって、再帰的イベントリスナによれば、階層データ構造を有するアプリケーションにおける現行の非再帰的イベントリスナに対して、少なくとも時間、メモリ、およびネットワーク帯域幅の消費の点で、システムの全体性能を大幅に改善可能である。

【0039】

図2は、開示技術の一実施形態に係る、階層データ構造のノードにおける再帰的イベントリスナを示したブロック図である。システム200は、図1の環境100または類似の環境とすることができる。システム200は、サーバ215と、通信ネットワーク210を介して当該サーバ215と通信する（communicating）クライアント205とを具備する。クライアント205およびサーバ215は、同じ物理的マシン（physical machine）または異なるマシン（machine）上で動作していてもよい。クライアント205は、サーバ215中のツリー構造220への再帰的イベントリスナ245の登録を要求することができる。たとえば、クライアント205は、ツリー構造220のノードBへの再帰的イベントリスナ245の登録を要求する。サーバ215と通信するように構成された受信モジュール225は、クライアント205から、ツリー構造220のノードBに再帰的イベントリスナ245を登録する要求を受信する。

【0040】

登録モジュール230は、ツリー構造220のノードBに再帰的イベントリスナ245を登録する。さらに、登録モジュール230は、サーバのメモリ中のノードBに対応するフィールド（たとえば、ビットまたはビットセット）を所定値に設定することによって、ノードBに再帰的イベントリスナ245が登録された旨を示す。ノードBに登録された再帰的イベントリスナ245は、ノードBにおけるイベントの発生をクライアント205に通知する。ノードBへの再帰的イベントリスナ245の登録後、階層判定モジュール235は、ノードBがツリー構造220に子孫ノードを有するか否かを判定する。ノードBが子孫ノードEおよびFを有する旨の判定に回答して、階層イベント処理モジュール240は、ノードBに加えてノードEまたはFにおけるイベントの発生を受信して（listen）通知するように、ノードBに登録された再帰的イベントリスナ245を設定する。再帰的イベントリスナ245は、ツリー構造220のノードB、E、またはFのうち少なくとも1つにおけるイベントの発生をクライアント205に通知する。したがって、再帰的イベントリスナ245によれば、単一のノードにのみ登録するだけで、複数のノードにおいて発生しているイベントの通知が得られるという利点がある。

【0041】

ツリー構造220のノードにおいて発生し得るイベントの種類としては、新たなノードの作成、既存ノードの修正、または既存ノードの消去が挙げられる。ノードB、E、またはFにおいて発生するイベントは、同じ種類であってもよいし、異なる種類であってもよい。さらに、イベントリスナ245は、上記イベント種類のいずれかの発生をクライアントに通知するように構成されている。

【 0 0 4 2 】

一実施形態において、ツリー構造 2 2 0 のノードはそれぞれ、データベース 2 5 0 等の記憶ユニットの異なる論理区画を表す。別の実施形態において、ツリー構造 2 2 0 のノードの少なくとも一部は、データベース 2 5 0 の異なる論理区画に含まれるデータを表す。たとえば、ツリー構造 2 2 0 は、様々なユーザのユーザプロファイルデータを多くのシャードに有するユーザデータベースを表していてもよい。さらに別の実施形態において、ノードは、システム 2 0 0 の他のエンティティを表していてもよい。ツリー構造 2 2 0 は、サーバ 2 1 5 のメモリに含まれていてもよい。

【 0 0 4 3 】

各モジュールすなわち受信モジュール 2 2 5、登録モジュール 2 3 0、階層判定モジュール 2 3 5、階層イベント処理モジュール 2 4 0、および再帰的イベントリスナ 2 4 5 はそれぞれ、サーバ 2 1 5 と同じマシンまたは異なるマシン上で動作可能であるか、サーバ 2 1 5 と統合可能であるか、またはサーバ 2 1 5 と協働して動作可能である。

【 0 0 4 4 】

また、クライアント 2 0 5 は、ノード A ~ G のいずれかに非再帰的イベントリスナを登録するようにしてもよい。一実施形態において、ノードのイベントを再帰的イベントリスナおよび非再帰的イベントリスナの両者が受信している場合は、イベントの発生の通知が 1 つだけクライアント 2 0 5 に送信される。たとえば、クライアント 2 0 5 は、ノード B に再帰的イベントリスナ 2 4 5 を登録し、ノード B またはその子孫ノード E および F に非再帰的イベントリスナ（図示せず）を登録している場合、ノード B、E、または F におけるイベントの発生時には、ノード B、E、または F のいずれかにおけるイベントの発生に関する通知のみを受信することになる。クライアントは、再帰的イベントリスナおよび非再帰的イベントリスナに対する別個の通知をサーバから受信していなくてもよい。

【 0 0 4 5 】

ノード、たとえばノード B に登録された再帰的イベントリスナ 2 4 5 は、ノード B、E、または F におけるイベントの発生がクライアント 2 0 5 に通知された後でも保持される。一方、ノードに登録された非再帰的イベントリスナは、当該ノードにおけるイベントの発生がクライアント 2 0 5 に通知された後に削除される。ノードにおけるイベントの発生の通知を継続して得るには、非再帰的イベントリスナを再度登録する必要がある。再帰的イベントリスナ 2 4 5 は、非再帰的イベントリスナとは異なり、ノードにおけるイベントの発生の通知を継続して得るために再度ノードに登録する必要がない。したがって、再帰的イベントリスナ 2 4 5 は、非再帰的イベントリスナと比べて、少なくともノードへの当該再帰的イベントリスナの登録に消費される時間およびネットワーク帯域幅の点で効率的である。

【 0 0 4 6 】

階層データ構造のノードに再帰的イベントリスナを登録する方法

図 3 は、開示技術の一実施形態に係る、階層データ構造のノードに再帰的イベントリスナを登録するプロセス 3 0 0 のフローチャートである。プロセス 3 0 0 は、図 2 のシステム 2 0 0 等のシステムに実装してもよい。ステップ 3 0 5 において、サーバ（またはサーバと協働する受信モジュール）は、クライアントから、起点ノードにイベントリスナを登録する要求を受信する。起点ノードは、階層中で互に関連する複数のノードのうちの 1 つである。階層中のノードはそれぞれ、データベースの異なる論理区画を表す。ステップ 3 1 0 において、サーバ（またはサーバと協働する登録モジュール）は、起点ノードにイベントリスナを登録し、起点ノードにおける第 1 のイベントの発生をクライアントに通知するようにイベントリスナを構成する。

【 0 0 4 7 】

判定ステップ 3 1 5 において、サーバ（またはサーバと協働する階層判定モジュール）は、起点ノードが階層中に子孫ノードを有するか否かを判定する。起点ノードが階層中に子孫ノードを有さない旨の判定に回答して、プロセス 3 0 0 は、ステップ 3 3 0 で復帰する。一方、起点ノードが階層中に子孫ノードを有する旨の判定に回答し、ステップ 3 2 0

10

20

30

40

50

において、サーバ（またはサーバと協働する階層イベント処理モジュール）は、子孫ノードに別のイベントリスナを登録する必要なく、子孫ノードにおける第2のイベントの発生をクライアントに通知するようにイベントリスナを設定する。

【0048】

起点ノードにおける第1のイベントまたは子孫ノードにおける第2のイベントの発生時に、イベントリスナは、ステップ325に示すように、第1のイベントまたは第2のイベントの少なくとも一方の発生をクライアントに通知する。

【0049】

図4は、開示技術の一実施形態に係る、階層データ構造のノードにおけるイベントの発生を通知するプロセス400のフローチャートである。プロセス400は、図2のシステム200等のシステムに実装してもよい。ステップ405において、サーバは、当該サーバ中のノードにおいてイベントが発生したか否かを判定する。ノードは、階層中で互いに関連する複数のノードのうちの1つである。ノードにおいてイベントが発生していない旨の判定に回答して、プロセスは、ステップ435で復帰する。

【0050】

一方、ノードにおいてイベントが発生した旨の判定に回答し、判定ステップ410において、サーバは、クライアントによってノードにイベントリスナが登録されたか否かを判定する。ノードにイベントリスナが登録された旨の判定に回答し、ステップ430において、イベントリスナは、当該ノードにおけるイベントの発生をクライアントに通知する。一方、ノードにイベントリスナが登録されていない旨の判定に回答し、判定ステップ415において、サーバは、ノードが階層中に先祖ノード（ascendant node）を有するか否かを判定する。ノードが階層中に先祖ノードを有さない旨の判定に回答して、プロセス400は、ステップ435で復帰する。

【0051】

一方、ノードが階層中に先祖ノードを有する旨の判定に回答し、判定ステップ420において、サーバは、先祖ノードにイベントリスナが登録されたか否かを判定する。先祖ノードにイベントリスナが登録された旨の判定に回答し、ステップ430において、イベントリスナは、当該ノードにおけるイベントの発生をクライアントに通知する。一方、先祖ノードにイベントリスナが登録されていない旨の判定に回答し、判定ステップ425において、サーバは、先祖ノードが階層のルートノードであるか否かを判定する。先祖ノードが階層のルートノードでない旨の判定に回答して、プロセスの制御はステップ415に移る。一方、先祖ノードが階層のルートノードである旨の判定に回答して、サーバは、当該ノードまたは先祖ノードに再帰的イベントリスナが登録されていないことを把握し、プロセス400は、ステップ435で復帰する。

【0052】

以上のように、再帰的イベントリスナによれば、1つのノードにのみ登録するだけで、複数のノードすなわち再帰的イベントリスナが登録されたノードおよび当該ノードの子孫ノードにおけるイベントの発生を通知するという利益がもたらされる。

【0053】

動作例

図5は、開示技術の一実施形態に係る、図1に示すようなコーディネーションサービスにおける再帰的イベントリスナの使用の一例を示している。コーディネーションサービス500は、ユーザデータベース510から得られたユーザプロファイルデータを階層データ構造515として表すサーバ505を具備する。データベース510は、多くのユーザのデータ、たとえばソーシャルネットワーキングアプリケーションにおけるユーザのデータを含んでいてもよい。また、データベース510は、多くの論理区画すなわちLP1、LP2から、以下同様にLPnまでを含む。論理区画はそれぞれ、ユーザデータベース510からの設定可能なユーザ数のデータを含む。階層データ構造515においては、ユーザデータベース510をルートノードと考えてもよく、また、ユーザデータベース510の論理区画をルートノードの子孫ノードと考えてもよい。

【 0 0 5 4 】

クライアントは、階層データ構造 5 1 5 のノードのいずれかに再帰的イベントリスナ、非再帰的イベントリスナ、またはその両者を登録するよう要求してもよい。たとえば、クライアント 5 2 5 がルートノードであるユーザデータベース 5 1 0 に登録された再帰的イベントリスナを有し、クライアント 5 3 0 が子孫ノード L P 2 に登録された再帰的イベントリスナおよび非再帰的イベントリスナを有し、クライアント 5 3 5 がルートノードであるユーザデータベース 5 1 0 に登録された非再帰的イベントリスナを有するシナリオを考える。すべてのイベントリスナすなわち再帰的および非再帰的イベントリスナは、階層データ構造 5 1 5 で発生している作成、消去、または修正等の様々なイベント種類のいずれかの発生をクライアントに通知可能である。

10

【 0 0 5 5 】

たとえば、L P_{n+1} 5 2 0 等の新たな論理区画の作成が階層データ構造 5 1 5 で発生しているものと仮定する。クライアント 5 2 5 は、この新たな子孫ノード L P_{n+1} 5 2 0 の作成の通知を受けることになる。ルートノードであるユーザデータベース 5 1 0 および子孫ノードにおけるイベントの発生をクライアント 5 2 5 の再帰的イベントリスナが受信するためである。クライアント 5 3 0 に関して、当該クライアント 5 3 0 により子孫ノード L P 2 に登録されたイベントリスナが再帰的であるか非再帰的であるかに関わらず、クライアント 5 3 0 は、子孫ノード 5 2 0 の作成の通知を受けることはない。子孫ノード 5 2 0 の作成は、ノード L P 2 または当該ノード L P 2 の子孫ノードで発生したイベントではないためである。クライアント 5 3 5 に関して、ルートノード 5 1 0 に登録された非再帰的イベントリスナは、ノード 5 1 0 においてのみ発生しているイベントを受信するため、新たな子孫ノード 5 2 0 の作成を把握しない。したがって、クライアント 5 3 5 は、階層データ構造における新たな子孫ノード 5 2 0 の作成の通知を受けることはない。

20

【 0 0 5 6 】

サーバ 5 0 5 は、再帰的イベントリスナの登録のためのアクセス制御機能を提供可能である。たとえば、ルートノードへの再帰的イベントリスナの登録は、ユーザセットまたはユーザ役割セット、たとえば管理者役割または監督者役割等に制限されていてもよい。また、アクセス制御機能によって、(i) 任意のノードへの再帰的イベントリスナの登録、(i i) 特定のイベント種類に対する再帰的イベントリスナの登録、(i i i) 階層中の特定経路に沿った再帰的イベントリスナの登録等を許可可能なユーザセットまたはユーザ役割セットを指定するようにしてもよい。

30

【 0 0 5 7 】

さらに、サーバは、再帰的イベントリスナ用のフィルタを提供するようにしてもよい。フィルタは、イベントの種類等を受信するようにイベントリスナを設定可能な階層データ構造中の経路を指定するものであってもよい。たとえば、図 2 のツリー構造 2 2 0 において、ルートノード A に登録された再帰的イベントリスナにフィルタを適用することにより、経路 A - B - F に沿ったノードにおいてのみ発生しているイベントを受信するようにしてもよい。別の例として、フィルタは、消去の種類のイベントの発生のみが通知されるように指定することも可能である。

40

【 0 0 5 8 】

一部の実施形態においては、ネットワーク分割、クライアントまたはサーバ障害等の障害によって、クライアントがサーバから切断される可能性がある。サーバは、当該クライアントに関連する再帰的イベントリスナの登録を自動的に取り消し、クライアントは、サーバからのイベント通知の受信を停止することになる。クライアントは、新たなサーバへの再帰的イベントリスナの再登録を試みて、切断中に発生したイベント通知を含め、イベント通知の受信を継続可能である。このように、クライアントは、所与のノードの階層で発生するすべてのイベントを受信するものと仮定できるとともに、クライアント側のアプリケーションにより保持されるノード構造がサーバ側と整合するものと仮定できる。

【 0 0 5 9 】

このメカニズムがなければ、クライアントは、再接続後、再帰的イベントリスナの登録

50

前に、階層全体の読み出しが必要となる場合がある。障害中にノード構造が変化してしまっている場合があるためである。少なくとも一部の実施形態においては、基本データベースで行われたノード階層の各変更（たとえば、作成、修正、消去）に処理番号が付与される。クライアントは、サーバからノードを読み出すたびに、ノードを変更した最新の処理番号も受信する。クライアントは、再接続を試みる場合、処理番号を提供することによって、ノードに再帰的イベントリスナを登録可能である。サーバは、提供された処理番号に対応するイベントの発生以降に発生したすべてのイベントの通知を試みる。

【0060】

サーバは、ノード階層を移動して、クライアントに送る必要があるイベントを確定する。また、サーバは、イベントの履歴（たとえば、データベースにログとして格納）を参照することによって、消去されたノードの一覧を確定する。消去されたノードは、所与のノードの現在の階層を移動しても確定できないためである。少なくとも一部の実施形態において、クライアントに送信されたイベントの一覧は、付随する処理番号に従ってソートされる。

10

【0061】

図6は、開示技術の一実施形態に係る、クライアントとサーバとの間の通信障害に応答して、クライアントにより再帰的イベントリスナを再登録するプロセス600を示したフローチャートである。プロセス600は、システム200等のシステムにおいて実行可能である。判定ステップ605において、サーバは、所与の処理番号が所定の閾時刻（threshold time）よりも古いかなかを判定する。少なくとも一部の実施形態において、サーバは、ある所定期間にわたってイベントの履歴を格納する。サーバは、処理が所定の閾時刻よりも古いと判定した場合、ステップ610において、再帰的イベントリスナを登録する要求を拒絶する。少なくとも一部の実施形態において、クライアントは、要求から処理番号を省くことにより、登録を再度試みることができる。

20

【0062】

一方、サーバは、処理が所定の閾時刻よりも古くないと判定した場合、判定ステップ615において、所与のノードの階層が未訪問ノード（unvisited node）すなわち消去されたノードを有するか否かを判定する。未訪問ノードが存在する場合、ステップ645において、サーバは、未訪問ノードそれぞれに対する消去イベントをイベントの一覧に追加する。少なくとも一部の実施形態において、サーバは、データベースに格納されたイベントのログ（処理の履歴）を読み出すことによって、消去されたノードを確定するようにしてもよい。一方、サーバは、消去されたノードが存在しないと判定した場合、判定ステップ620において、現在のノードが（所与の処理番号に対応する）処理の発生後に作成されたか否かを判定する。

30

【0063】

サーバは、現在のノードが処理後に追加されたと判定した場合、ステップ625において、ノード作成イベントを一覧に追加して、制御をステップ640に受け渡す。一方、サーバは、現在のノードが処理後に追加されたものではないと判定した場合、判定ステップ630において、現在のノードが処理後に修正されたか否かを判定する。サーバは、現在のノードが処理後に修正された旨を判定した場合、ステップ635において、ノード修正イベントを一覧に追加して、制御をステップ640に受け渡す。ステップ640において、サーバは、階層中の次の子孫ノードに移動する。少なくとも一部の実施形態において、サーバは、横型探索の順序で階層を移動する。サーバは、所与のノードの階層全体を移動したら、ステップ650において、イベントの一覧を付随するイベント処理番号に従ってソートし、ステップ655において、イベントの一覧をクライアントに送る。

40

【0064】

実装のための装置

図7は、開示技術に係る、様々な動作を実行するとともに、このような動作で生成および使用される、あるいはこのような動作で生成または使用される、様々な情報を格納可能な装置のブロック図である。これらの装置は、本明細書に記載の任意のコンピュータを表

50

することができる。コンピュータ700は、コーディネーションサービス、サーバ、クライアント、データベース、ツリー構造等、図1～図6の例に示したエンティティ、構成要素、またはサービス（および本明細書に記載のその他任意の構成要素）のいずれかを実装可能なハードウェア装置を示すためのものである。コンピュータ700は、1つまたは複数のプロセッサ701と、相互接続703に結合されたメモリ702とを具備する。図7において、相互接続703は、適当なブリッジ、アダプタ、またはコントローラによって接続された任意の1つまたは複数の独立した物理的バス、ポイントツーポイント接続、またはその両者を表す抽象化概念として示されている。したがって、相互接続703は、たとえばシステムバス、PCI（Peripheral Component Interconnect）バスもしくはPCIエクスプレスバス、ハイパートランスポートもしくはISA（Industry Standard Architecture）バス、SCSI（Small Computer System Interface）バス、USB（Universal Serial Bus）、IIC（I2C）バス、または「ファイアワイア」とも称するIEEE（Institute of Electrical and Electronics Engineers）規格1394バスを含んでいてもよい。

10

【0065】

プロセッサ701は、コンピュータ700の中央演算処理装置（CPU）であるため、コンピュータ700の全体的な動作を制御する。特定の実施形態において、プロセッサ701は、メモリ702に格納されたソフトウェアまたはファームウェアを実行することによってこれを実現する。プロセッサ701は、1つまたは複数のプログラム可能な汎用または専用マイクロプロセッサ、デジタルシグナルプロセッサ（DSP）、プログラマブルコントローラ、特定用途向け集積回路（ASIC：Application Specific Integrated Circuit）、プログラマブルロジックデバイス（PLD）、トラステッドプラットフォームモジュール（TPM）等、またはこのようなデバイスの組み合わせであってもよいし、これらを含んでいてもよい。

20

【0066】

メモリ702は、コンピュータ700のメインメモリであるか、またはメインメモリを含む。メモリ702は、ランダムアクセスメモリ（RAM）、リードオンリーメモリ（ROM）、フラッシュメモリ等、またはこのようなデバイスの組み合わせのうち、任意の形態を表す。使用に際して、メモリ702は、コードを含んでいてもよい。一実施形態において、コードは、コンピュータバスインタフェースを介して受信した汎用プログラムを認識し、当該汎用プログラムをプロセッサでの実行に備えるように構成された一般的なプログラミングモジュールを含む。別の実施形態において、この一般的なプログラミングモジュールは、ASIC、PLD（Programmable Logic Device）、またはフィールドプログラマブルゲートアレイ（FPGA）等のハードウェア回路を用いて実装してもよい。

30

【0067】

プロセッサ701には、相互接続703を介して、ネットワークアダプタ707、記憶装置705、およびI/O装置706も接続されている。ネットワークアダプタ707は、コンピュータ700のネットワークを介した遠隔装置との通信を可能とし、たとえばイーサネットアダプタまたはファイバチャネルアダプタであってもよい。また、ネットワークアダプタ707は、コンピュータ700とそのクラスタ内の他のコンピュータとの通信を可能にするものであってもよい。一部の実施形態において、コンピュータ700は、2つ以上のネットワークアダプタを用いることにより、クラスタ内外の通信を別個に扱うようにしてもよい。

40

【0068】

I/O装置706としては、たとえばキーボード、マウス等のポインティングデバイス、ディスクドライブ、プリンタ、スキャナ、および表示装置等の他の入力および出力装置、あるいはいずれか一方が挙げられる。表示装置としては、たとえば陰極線管（CRT：

50

Cathode Ray Tube)、液晶ディスプレイ(LCD:Liquid Crystal Display)、またはその他何らかの適用可能な既知または従来の表示装置が挙げられる。

【0069】

メモリ702に格納されたコードは、ソフトウェアおよびファームウェア、あるいはいずれか一方として実装することにより、上記動作を実行するようにプロセッサ701をプログラミング可能である。特定の実施形態において、このようなソフトウェアまたはファームウェアは最初、コンピュータ700を通して(たとえば、ネットワークアダプタ707を介して)遠隔システムからダウンロードすることによりコンピュータ700に提供するようにしてもよい。

10

【0070】

本明細書に導入した技術は、たとえばソフトウェアおよびファームウェア、あるいはいずれか一方でプログラムされたプログラム可能な回路(たとえば、1つまたは複数のマイクロプロセッサ)による実装、専用ハードワイヤード(プログラム不可能な)回路における全体実装、またはこれらの形態の組み合わせにおける実装が可能である。専用ハードワイヤード回路は、たとえば1つまたは複数のASIC、PLD、FPGA等の形態であってもよい。

【0071】

本明細書に導入した技術の実装に用いるソフトウェアまたはファームウェアは、マシン可読記憶媒体に格納されていてもよく、また、1つまたは複数の汎用または専用のプログラム可能なマイクロプロセッサにより実行するようにしてもよい。本明細書において使用する用語としての「マシン可読記憶媒体」は、マシンによりアクセス可能な形態の情報を格納可能な任意のメカニズムを含む。

20

【0072】

また、マシンは、サーバコンピュータ、クライアントコンピュータ、パーソナルコンピュータ(PC)、タブレットPC、ラップトップコンピュータ、セットトップボックス(STB)、個人用デジタル補助装置(PDA:Personal Digital Assistant)、携帯電話、iフォン、ブラックベリー、プロセッサ、電話、ウェブ機器、ネットワークルータ、スイッチもしくはブリッジ、または行う動作を指定する命令セット(シーケンシャルまたはそれ以外)を実行可能な任意のマシンとすることができる。

30

【0073】

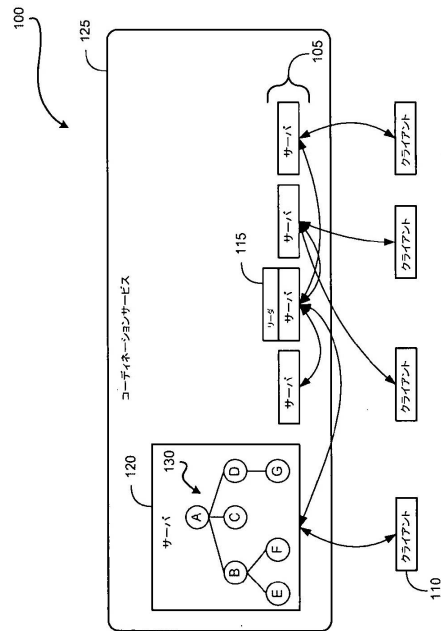
マシンアクセス可能な記憶媒体または記憶装置705としては、たとえば記録可能/記録不可能な媒体(たとえば、ROM、RAM、磁気ディスク記憶媒体、光記憶媒体、フラッシュメモリデバイス等)またはそれらの任意の組み合わせが挙げられる。記憶媒体は通常、持続性であってもよいし、持続性デバイスを含んでいてもよい。これに関連して、持続性記憶媒体は、物理的状态が変化し得る有形すなわち明確な物理的形態のデバイスを含んでいてもよい。したがって、「持続性」とは、たとえば状態の変化に関わらず有形性を維持するデバイスを表す。

【0074】

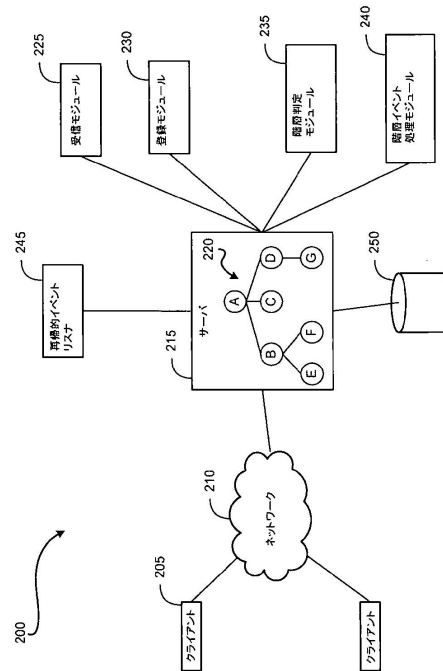
本明細書において、用語「ロジック」は、たとえば特定のソフトウェアおよびファームウェア、あるいはいずれか一方でプログラムされたプログラム可能な回路、専用ハードワイヤード回路、またはそれらの組み合わせを含むことができる。

40

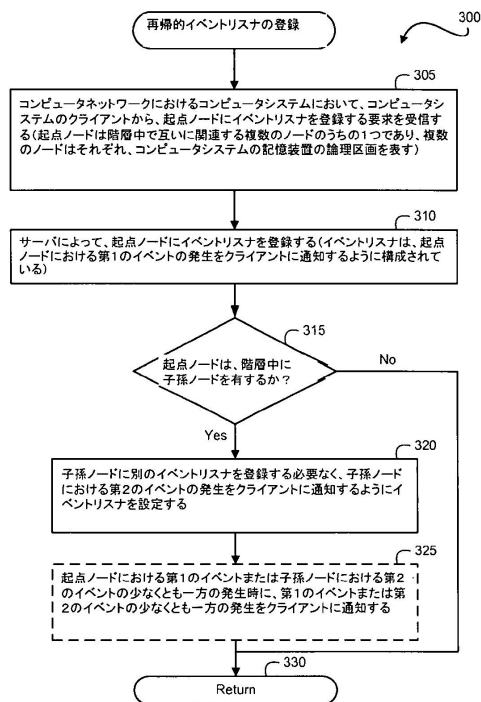
【図 1】



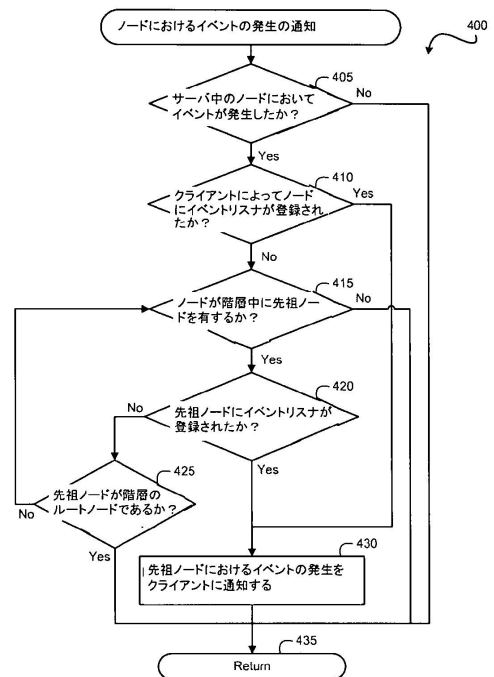
【図 2】



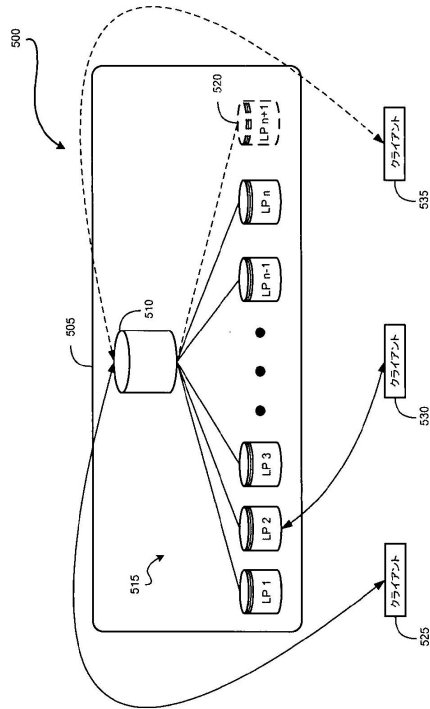
【図 3】



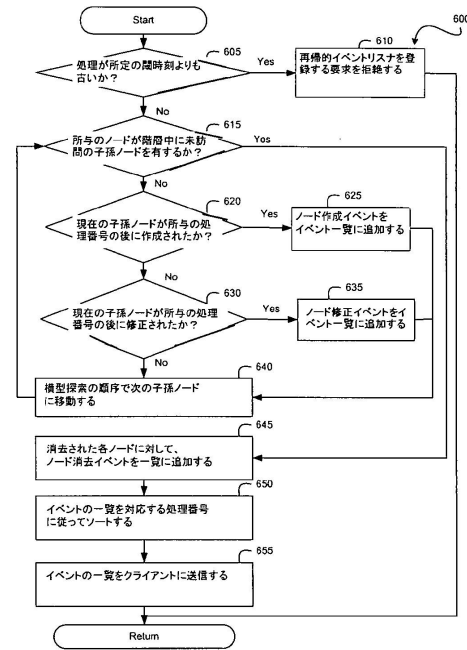
【図 4】



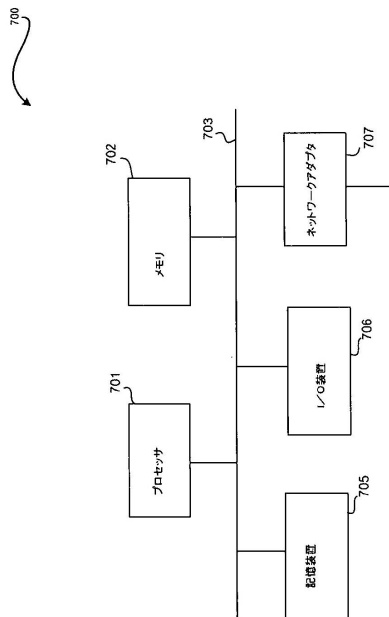
【図 5】



【図 6】



【図 7】



フロントページの続き

(72)発明者 カスリア、ビシャル

アメリカ合衆国 94025 カリフォルニア州 メンロー パーク ウィロー ロード 160
1

審査官 田中 幸雄

(56)参考文献 特表2002-535792(JP, A)

国際公開第2011/099649(WO, A1)

特開2007-234014(JP, A)

特表2013-524360(JP, A)

特開2007-52563(JP, A)

(58)調査した分野(Int.Cl., DB名)

G06F 9/54

G06F 9/48