(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2016-519379
(P2016-519379A)
(43)公表日　平成28年6月30日(2016.6.30)

| (51)Int.Cl. | | F I | | | テーマコード（参考） |
|---|---|---|---|---|---|
| G06F 12/00 | (2006.01) | G06F | 12/00 | 535F | |
| | | G06F | 12/00 | 518A | |
| | | G06F | 12/00 | 545A | |

審査請求 有　予備審査請求 未請求　（全 35 頁）

| | |
|---|---|
| (21)出願番号 | 特願2016-514043 (P2016-514043) |
| (86)(22)出願日 | 平成26年5月13日 (2014.5.13) |
| (85)翻訳文提出日 | 平成28年1月7日 (2016.1.7) |
| (86)国際出願番号 | PCT/US2014/037901 |
| (87)国際公開番号 | WO2014/186396 |
| (87)国際公開日 | 平成26年11月20日 (2014.11.20) |
| (31)優先権主張番号 | 13/893,004 |
| (32)優先日 | 平成25年5月13日 (2013.5.13) |
| (33)優先権主張国 | 米国 (US) |

(71)出願人　507303550
アマゾン・テクノロジーズ・インコーポレーテッド
アメリカ合衆国・８９５０７・ネバダ州・レノ・ピーオーボックス　８１０２
(74)代理人　100064621
弁理士　山川　政樹
(74)代理人　100098394
弁理士　山川　茂樹
(72)発明者　バーチャル，ローリオン・ダレル
アメリカ合衆国・９８１０９－５２１０・ワシントン州・シアトル・テリー　アヴェニュ　ノース・４１０

最終頁に続く

(54)【発明の名称】トランザクションの順序付け

(57)【要約】
データベースサービスのノードは、データベースサービスによって記憶されたレコードの読み取りを実行する読み取り要求、及びそのレコードに対してトランザクションを実行するトランザクション要求を受け取ることができる。第１の時刻指示と第２の時刻指示とを、読み取りとトランザクションとにそれぞれ関連付けることができる。潜在的な読み取り異常（例えば、ファジーリード、リードスキューなど）は、第１の時刻指示が、第２の時刻指示の閾値内にあると判定したことに少なくとも部分的に基づいて検出され得る。潜在的な読み取り異常を検出したことに応答して、第１の時刻指示が第２の時刻指示よりも早い時点を示しているかどうかに関わらず、トランザクション要求によって指定されたトランザクションの後に上記読み取りが実行され得る。
【選択図】図３

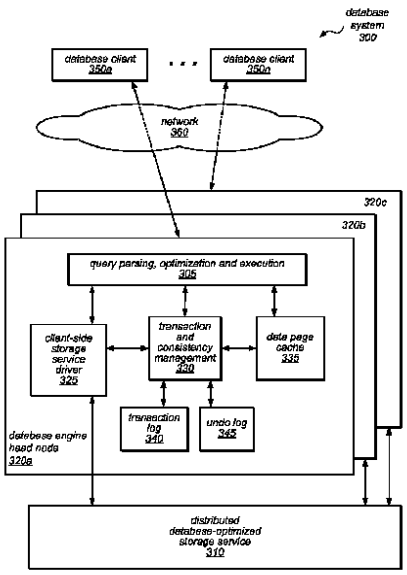



FIG. 3

【特許請求の範囲】
【請求項１】
システムであって、
それぞれが少なくとも１つのプロセッサ、及びメモリを含む複数のコンピューティングノードであって、データベースサービスを集合的に実施するように構成されている複数のコンピューティングノード、を含み、
前記複数のコンピューティングノードの第１のノードは、
データベーステーブル内の特定のデータページの特定のデータレコードを対象とする書き込み要求であって、前記特定のデータレコードに対して行うべき変更を指定する書き込み要求を受け取り、
前記変更をコミットする時刻を示すコミット時刻を前記書き込み要求に割り当てるように構成されており、
前記複数のコンピューティングノードの第２のノードは、
前記データベーステーブル内の前記特定のデータページを読み取る読み取り要求を受け取り、
読み取りの一貫性ポイントを示す読み取り一貫性ポイント時刻を前記読み取り要求に割り当て、
前記読み取り一貫性ポイント時刻が前記コミット時刻の精度ウィンドウ内にあると判定し、
前記読み取り一貫性ポイント時刻が前記コミット時刻よりも早い時点を示しているかどうかに関わらず、前記書き込み要求によって指定された前記変更がコミットされた後に前記読み取り要求の実行を行わせるように構成されている、システム。
【請求項２】
前記複数のコンピューティングノードの前記第２のノードは、
前記書き込み要求に関連付けられた書き込みアンラッチ時刻が、前記読み取り要求に関連付けられた読み取りラッチ時刻の精度ウィンドウ内にあると判定するようにさらに構成されており、
前記書き込み要求によって指定された前記変更がコミットされた後に前記読み取り要求の実行を前記行わせることは、
前記読み取りラッチ時刻を後の時点に移動すること、及び
前記読み取り一貫性の時点以降に前記読み取り要求を再試行すること、を含む請求項１に記載のシステム。
【請求項３】
前記書き込み要求によって指定された前記変更がコミットされた後に前記読み取り要求の実行を前記行わせることは、前記移動された読み取りラッチ時刻が前記書き込みアンラッチ時刻の精度ウィンドウ内にはないと判定すること、をさらに含む、請求項２に記載のシステム。
【請求項４】
前記第１のノード及び前記第２のノードのそれぞれは、互いの前記精度ウィンドウ内に維持されている個別のクロックを保持するように構成されており、前記コミット時刻が前記第１のノードによって決定され、前記読み取り一貫性の時点が前記第２のノードによって決定される、請求項１に記載のシステム。
【請求項５】
方法であって、
複数のコンピューティングノードによって、
１つ以上のクライアントから、記憶されたレコードの読み取りを実行する読み取り要求、及び前記レコードの更新を実行する更新要求を受け取ること、
第１の時刻指示と第２の時刻指示とを、前記読み取りと前記更新とにそれぞれ関連付けること、並びに
前記第１の時刻指示が前記第２の時刻指示の閾値内にあると判定したことに少なくとも

部分的に基づき、潜在的な読み取り異常を検出すること、
を実行すること、を含む方法。
【請求項６】
前記潜在的な読み取り異常を前記検出したことに応答して、前記第１の時刻指示が前記第２の時刻指示よりも早い時点を示しているかどうかに関わらず、前記更新要求によって指定された前記更新の後に前記読み取り要求によって指定された前記読み取りを実行させること、をさらに含む、請求項５に記載の方法。
【請求項７】
前記読み取りの再試行に関連付けられた再試行の時刻指示が前記第１の時刻指示よりも後の時刻を示すように、前記第１の時刻指示以降に前記読み取りを再試行すること、及び
前記再試行された読み取りに対して読み取り異常が発生していないと判定すること、をさらに含む請求項５に記載の方法。
【請求項８】
前記第２の時刻指示は前記更新をコミットする時刻を示す、請求項５に記載の方法。
【請求項９】
前記潜在的な読み取り異常を前記検出することは、第３の時刻指示にさらに基づいており、前記第３の時刻指示も前記更新要求に関連付けられている、請求項５に記載の方法。
【請求項１０】
前記潜在的な読み取り異常を前記検出することは、前記第３の時刻指示が第４の時刻指示の閾値内にあると判定したことにさらに基づいており、前記第４の時刻指示も前記読み取り要求に関連付けられている、請求項９に記載の方法。
【請求項１１】
前記読み取り要求は前記複数のノードの第１のノードによって受け取られ、前記更新要求は前記複数のノードの別の第２のノードによって受け取られ、前記第１のノード及び前記第２のノードは個別のクロックをそれぞれ保持しており、前記読み取りに関連付けられた前記第１の時刻指示は前記第１のノードの前記個別のクロックによって決定され、前記更新に関連付けられた前記第２の時刻指示は前記第２のノードの前記個別のクロックによって決定される、請求項５に記載の方法。
【請求項１２】
前記潜在的な読み取り異常は潜在的なファジーリードである、請求項５に記載の方法。
【請求項１３】
再試行の頻度に少なくとも部分的に基づいて前記閾値を変更すること、をさらに含む、請求項５に記載の方法。
【請求項１４】
前記検出することは、前記レコードの前記読み取り、及び別のレコードの読み取りまたは前記レコードの第２の読み取りのいずれかを含む読み取り要求に対して実行される、請求項５に記載の方法。
【請求項１５】
システムであって、
１つ以上のプロセッサと、
プログラム命令が記憶された１つ以上のメモリであって、前記プログラム命令が、データベースサービスのデータベースノードを実装するために前記１つ以上のプロセッサによってコンピュータで実行可能である１つ以上のメモリと、を含み、前記データベースノードは、
前記データベースサービスによって記憶されたレコードの読み取りを指定する読み取り要求によって指定された前記読み取りに第１の時刻指示を関連付け、
前記第１の時刻指示が、別のデータベースノードによって受け取られたトランザクション要求であって、前記レコードを変更するトランザクションを指定するトランザクション要求によって指定された前記トランザクションに関連付けられた第２の時刻指示の閾値内にあると判定し、

前記第１の時刻指示が前記第２の時刻指示よりも早い時点を示しているかどうかに関わらず、前記トランザクション要求によって指定された前記変更がコミットされた後に前記読み取り要求の実行を行わせるように構成されている、システム。

【発明の詳細な説明】

【背景技術】

【０００１】

ソフトウェアスタックの様々な構成要素を分散させることにより、ある場合には、（例えば、複製による）耐障害性、高度な持続性、及び（例えば、大型で高価な構成要素を少数使用するのではなく、それよりも小型で安価な構成要素を多数使用することによって）費用がより少なくて済む解決策が得られる（またはそれらの助けになる）。しかしながら、データベースは、歴史的に見ると、分散に最も適用し難いソフトウェアスタックの構成要素間に存在してきた。例えば、データベースを分散させる一方で、データベースによる提供が期待されるいわゆるＡＣＩＤ特性（例えば、原子性（Ａｔｏｍｉｃｉｔｙ）、一貫性（Ｃｏｎｓｉｓｔｅｎｃｙ）、独立性（Ｉｓｏｌａｔｉｏｎ）及び持続性（Ｄｕｒａｂｉｌｉｔｙ））を依然として保証することは困難な場合がある。特に、一貫性及び独立性の各特性に関しては、分散データベースシステムのノード間を整合してノード全体の因果関係を保つことは、従来のシステムにとって非常に難しいことが証明されている。

【図面の簡単な説明】

【０００２】

【図１】一実施形態に係る、データベースソフトウェアスタックの様々な構成要素を示すブロック図である。

【図２】いくつかの実施形態に係る、トランザクションの順序付けを行うように構成されたＷｅｂサービスベースのデータベースサービスを実装するように構成され得るサービスシステムアーキテクチャを示すブロック図である。

【図３】一実施形態に係る、トランザクションの順序付けを行うように構成されたデータベースシステムの様々な構成要素を示すブロック図である。

【図４】一実施形態に係る、トランザクションの順序付けを行うように構成された分散データベース最適化ストレージシステムを示すブロック図である。

【図５】一実施形態に係る、トランザクションの順序付けを行うように構成されたデータベースシステム内の別個の分散データベース最適化ストレージシステムの利用を示すブロック図である。

【図６】トランザクションの順序付け方法の一実施形態を示すフロー図である。

【図７】図７Ａ～７Ｃは、様々な実施形態に係る様々なトランザクションの順序付けを行うシナリオを示すタイミング図である。

【図８】様々な実施形態に係る、トランザクションの順序付けを実装するように構成されたコンピュータシステムを示すブロック図である。

【０００３】

本明細書では、いくつかの実施形態及び説明図の実施例として実施形態を説明するが、当業者であれば、記載の実施形態または図面にこれらの実施形態が限定されないことを認識するであろう。図面及びそれに対する詳細な説明は、開示された特定の形態に実施形態を限定することを意図するものではなく、むしろ、本発明は、添付された特許請求の範囲によって定められるような概念及び趣旨に該当する全ての修正形態、均等形態及び代替形態を包含するものであることを理解すべきである。本明細書で用いられる見出しは、体系化を目的とするにすぎず、本明細書または特許請求の範囲を限定するために用いられることを意味するものではない。本出願全体を通して用いられるように、用語「し得る（ｍａｙ）」は、義務的な意味（すなわち、～しなければならないを意味する）ではなく、許容的な意味（すなわち、～する可能性を有することを意味する）で用いられる。用語「含む（ｉｎｃｌｕｄｅ）」、「含んでいる（ｉｎｃｌｕｄｉｎｇ）」及び「含む（ｉｎｃｌｕｄｅｓ）」は、非制限的な関係を示しており、従って、含むがこれに限定されるものではないことを意味する。同様に、用語「有する（ｈａｖｅ）」、「有している（ｈａｖｉｎ

ｇ）」及び「有する（ｈａｓ）」も非制限的な関係を示すものであり、従って、有するがこれに限定されるものではないことを意味する。本明細書で用いられる用語「第１の（ｆｉｒｓｔ）」、「第２の（ｓｅｃｏｎｄ）」及び「第３の（ｔｈｉｒｄ）」などは、それらに先行する名詞のラベルとして用いられるものであり、順序付けなどが明示的に示されない限り、（例えば、空間的、時間的、論理的などの）いかなる順序付けも意味しない。

【０００４】

種々の構成要素は、１つ以上のタスクを実行「するように構成された（ｃｏｎｆｉｇｕｒｅｄ　ｔｏ）」と記載されることがある。このような文脈では、「するように構成された（ｃｏｎｆｉｇｕｒｅｄ　ｔｏ）」は、動作中に１つ以上のタスクを実行する「構造を備えている（ｈａｖｉｎｇ　ｓｔｒｕｃｔｕｒｅ　ｔｈａｔ）」ことを広く意味する広義の記述である。従って、構成要素がタスクを現在実行していないときでも、そのタスクを実行するようにその構成要素を構成することができる（例えば、コンピュータシステムは、現在動作が実行されていないときでも、その動作を実行するように構成され得る）。ある文脈では、「するように構成された（ｃｏｎｆｉｇｕｒｅｄ　ｔｏ）」は、動作中に１つ以上のタスクを実行する「回路を備えている（ｈａｖｉｎｇ　ｃｉｒｃｕｉｔｒｙ　ｔｈａｔ）」ことを広く意味する構造についての広義の記述であり得る。従って、構成要素が現在オンになっていないときでも、タスクを実行するようにその構成要素を構成することができる。通常、「するように構成された（ｃｏｎｆｉｇｕｒｅｄ　ｔｏ）」に相当する構造をなす回路は、ハードウェア回路を含み得る。

【０００５】

種々の構成要素は、本明細書において便宜上、１つ以上のタスクを実行するものとして説明され得る。こうした説明は、「するように構成された（ｃｏｎｆｉｇｕｒｅｄ　ｔｏ）」という語句を含むものとして解釈されるべきである。１つ以上のタスクを実行するように構成されている構成要素について述べることは、その構成要素に対して米国特許法第１１２条第６段落の解釈を援用することを明示的に意図していない。

【０００６】

「～に基づく（Ｂａｓｅｄ　Ｏｎ）」。本明細書で使用する場合、本用語は、決定に影響を及ぼす１つ以上の要因を説明するために用いられる。本用語は、決定に影響を及ぼし得る追加の要因を除外しない。すなわち、決定は、これらの要因だけに基づくものであってもよく、またはこれらの要因に少なくとも部分的に基づくものであってもよい。「Ｂに基づいてＡを決定する（ｄｅｔｅｒｍｉｎｅ　Ａ　ｂａｓｅｄ　ｏｎ　Ｂ）」という語句について考察する。Ｂは、Ａの決定に影響を及ぼす要因であり得るが、このような語句は、Ａの決定がＣにも基づいていることを除外しない。他の例では、ＡはＢのみに基づいて決定されてもよい。

【０００７】

本開示の範囲は、本明細書において取り組まれる問題のいずれかまたは全てを軽減するものであろうとなかろうと、（明示的または暗示的に）本明細書に開示された任意の特徴もしくは特徴の任意の組み合わせ、またはそれらの任意の一般化を包含する。従って、本出願（または本出願の優先権を主張する出願）の手続き中に、任意のこうした特徴の組み合わせに対して新たな特許請求の範囲を作成してもよい。特に、添付された特許請求の範囲を参照すると、従属請求項からの特徴は、独立請求項の特徴と組み合わせてもよく、各独立請求項からの特徴は、添付された特許請求の範囲に列挙された特定の組み合わせのみならず、任意の適切な方法で組み合わせてもよい。

【発明を実施するための形態】

【０００８】

トランザクションの順序付けを行う様々な実施形態を開示する。本実施形態の様々な実施形態は、記憶されたレコードの読み取りを実行する読み取り要求、及びそのレコードに対して（例えば、書き込みなどの）トランザクションを実行するトランザクション要求を受け取る（例えば、データベースサービスの）ノードを含み得る。本実施形態の様々な実施形態は、第１の時刻指示と第２の時刻指示とを、読み取りとトランザクションとにそれ

ぞれ関連付けるノードも含み得る。本実施形態の様々な実施形態は、第１の時刻指示が第２の時刻指示の閾値内にあると判定したことに少なくとも部分的に基づき、潜在的な読み取り異常（例えば、ファジーリード、リードスキューなど）を検出することをさらに含み得る。なお、いくつかの実施形態では、検出は、第１の時刻指示及び第２の時刻指示以外の時刻指示に基づくものであってもよい。潜在的な読み取り異常を検出したことに応答して、第１の時刻指示が第２の時刻指示よりも早い時点を示しているかどうかに関わらず、トランザクション要求によって指定されたトランザクションの後に上記読み取りが実行され得る。いくつかの事例では、読み取りを再試行することにより、その再試行については潜在的な読み取り異常が発生しないようにしてもよい。

【０００９】

まず、本明細書では、開示されたトランザクションの順序付け技術を実装するように構成された例示的なＷｅｂサービスベースのデータベースサービスについて説明する。この例示的なＷｅｂサービスベースのデータベースサービスに関する説明には、データベースエンジン及び別個の分散データベースストレージサービスなどの、例示的なＷｅｂサービスベースのデータベースサービスの様々な態様が含まれる（なお、いくつかの実施形態では、データベースエンジンから上記ストレージサービスを分けなくてもよい）。次いで、本明細書では、トランザクションの順序付け方法の様々な実施形態のフローチャートについて説明する。次に、本明細書では、開示された技術を実装可能な例示的なシステムについて説明する。本明細書全体を通じて、様々な実施例が提示される。なお、開示されたトランザクションの順序付け技術は、図１～５の例示的なデータベースサービス以外のシステム（データの読み取り、書き込み及び記憶に使用可能な他のシステムなど）において用いられてもよい。例えば、開示された技術は、データの読み取りとそのデータに対する一連の更新とが、上記読み取りからそれらの更新が見えるようになった時点で行われ得る任意のシステムにおいて用いられてもよい。

【００１０】

本明細書で説明されたシステムは、いくつかの実施形態では、クラウドコンピューティング環境においてクライアント（例えば、加入者）がデータストレージシステムを運用可能にするＷｅｂサービスを実装し得る。いくつかの実施形態では、データストレージシステムは、高度にスケーラブルかつ拡張可能な企業クラスのデータベースシステムであってよい。いくつかの実施形態では、複数の物理リソースにわたって分散されているデータベースストレージにクエリを送ってもよく、必要に応じてデータベースシステムをスケールアップまたはスケールダウンしてもよい。データベースシステムは、種々の実施形態では、様々な形式及び／または編成のデータベーススキーマを用いて有効に機能し得る。いくつかの実施形態では、クライアント／加入者は、いくつかの方法を用いて、例えば、ＳＱＬインターフェースを介して対話的に、データベースシステムにクエリを送出してもよい。他の実施形態では、外部のアプリケーション及びプログラムが、ＯＤＢＣ（Ｏｐｅｎ Ｄａｔａｂａｓｅ Ｃｏｎｎｅｃｔｉｖｉｔｙ）ドライバインターフェース及び／またはＪＤＢＣ（Ｊａｖａ Ｄａｔａｂａｓｅ Ｃｏｎｎｅｃｔｉｖｉｔｙ）ドライバインターフェースを用いてデータベースシステムにクエリを送出してもよい。

【００１１】

より具体的には、本明細書で説明されたシステムは、いくつかの実施形態では、１つのデータベースシステムの様々な機能要素が本質的に分散されているサービス指向データベースアーキテクチャを実装してもよい。例えば、複数の包括的かつ一体的なデータベースインスタンス（これらのインスタンスのそれぞれは、アプリケーションサーバ、検索機能、またはデータベースの基幹機能を提供するのに必要とされるもの以外の他の機能などの、外部機能を備え得る）を１つに結び付けるのではなく、これらのシステムは、データベースの基本動作（例えば、クエリ処理、トランザクション管理、キャッシング及びストレージ）を、個別的かつ独立的にスケーラブルな各層に編成してもよい。例えば、いくつかの実施形態では、本明細書で説明されたシステム内の各データベースインスタンスは、データベース階層（この階層は、１つのデータベースエンジン・ヘッドノード、及びクライ

アント側ストレージシステムドライバを含み得る）、並びに別個の分散ストレージシステム（このシステムは、既存システムのデータベース階層において従来行われた動作の一部を集合的に実行する複数のストレージノードを含み得る）を含んでもよい。本明細書で既に述べたように、説明されたトランザクションの順序付け技術は、他のシステムにおいても同様に適用され得る。

【００１２】

本明細書でより詳細に説明するように、いくつかの実施形態では、データベースの最下位レベルの動作の一部（例えば、バックアップ、復元、リカバリ、ログレコード操作及び／または様々な空き領域管理動作）をデータベースエンジンからストレージ層に肩代わりさせ、これらを、複数のノード及びストレージ装置にわたって分散させてもよい。例えば、いくつかの実施形態では、データベースエンジンが、データベーステーブル（またはそのデータページ）に変更を適用し、次いで変更されたデータページをストレージ層に送るのではなく、記憶されたデータベーステーブル（及びそのデータページ）に変更を適用することをストレージ層自体が担当してもよい。このような実施形態では、変更されたデータページではなくＲＥＤＯログレコードをストレージ層に送ってもよく、その後、ＲＥＤＯ処理（例えば、ＲＥＤＯログレコードの適用）を（例えば、バックグラウンドプロセスによって）ある程度緩慢に、かつ分散的な方法で実行してもよい。いくつかの実施形態では、クラッシュリカバリ（例えば、記憶されたＲＥＤＯログレコードからデータページを再構築すること）は、ストレージ層によって行われてもよく、分散された（かつ、場合によっては、緩慢な）バックグラウンドプロセスによって行われてもよい。

【００１３】

いくつかの実施形態では、ＲＥＤＯログ（及び未変更のデータページ）のみがストレージ層に送られるため、データベース階層とストレージ層の間のネットワークトラフィックが、既存のデータベースシステムにおけるネットワークトラフィックと比べて大幅に減少し得る。いくつかの実施形態では、各ＲＥＤＯログは、そのログが変更を指定するデータページに相当するサイズの１／１０程度となり得る。なお、データベース階層及び分散ストレージシステムから送られる要求は非同期であってよく、こうした複数のかかる要求を一度に移送してもよい。

【００１４】

一般に、１つのデータが与えられた上で、データベースにとって求められる最も重要なことは、データベースが最終的に同一のデータを戻すことができることである。これを行うために、データベースがいくつかの異なる構成要素（または階層）を含み、これらの構成要素のそれぞれが、異なる機能を実行するようにしてもよい。例えば、従来のデータベースは、３つの階層、すなわち、クエリを解析し、最適化し、実行する第１階層、トランザクション性、リカバリ及び持続性を提供する第２階層、並びにローカル接続のディスク上、またはネットワーク接続のストレージ上のいずれかにストレージを提供する第３階層を持っているものとみなすことができる。上述したように、従来のデータベースをスケーリングするための従来の試みでは、通常、上記データベースの３つの階層全てを複製すること、及びこれらの複製されたデータベースインスタンスを複数のマシンにわたって分散させることが求められている。

【００１５】

いくつかの実施形態では、本明細書で説明されたシステムは、スケーリングを実装するために、従来のデータベースを用いたシステムとは違ってデータベースシステムの機能を分割してもよく、（完全なデータベースインスタンスではなく）それらの機能要素のサブセットのみを複数のマシンにわたって分散させてもよい。例えば、いくつかの実施形態では、クライアントに直接対応する階層は、データの記憶手順または検索手順ではなく、どのデータを記憶または検索すべきかを指定する要求を受け取るように構成され得る。この階層が要求の解析及び／または最適化（例えば、ＳＱＬの解析及び最適化）を実行してもよく、それと共に別の階層がクエリを実行する役割を果たしてもよい。いくつかの実施形態では、第３階層は、トランザクション性及び結果の一貫性を提供する役割を果たしても

よい。例えば、この階層は、いわゆるＡＣＩＤ特性の一部、特に、データベースを対象とするトランザクションの原子性、データベース内の一貫性の維持、及びデータベースを対象とするトランザクション間の独立性を保証することを実現するように構成され得る。いくつかの実施形態では、この第３階層は、開示されたトランザクションの順序付け技術を実装してもよい。いくつかの実施形態では、次いで、第４階層が、様々な種類の障害の存在下で記憶データの持続性を提供する役割を果たしてもよい。例えば、この階層は、変更ロギング、データベースクラッシュからのリカバリ、下位のストレージ容量へのアクセスの管理、及び／または下位のストレージ容量における空き領域の管理を行う役割を果たしてもよい。

【００１６】

次に図面に目を向けると、図１は、一実施形態に係る、データベースソフトウェアスタックの様々な構成要素を示すブロック図である。本実施例に示すように、データベースインスタンスが複数の機能要素（または機能層）を含み、これらの機能要素のそれぞれが、データベースインスタンスの機能の一部を提供するようにしてもよい。本実施例では、データベースインスタンス１００は、（１１０に示す）クエリ解析・クエリ最適化層、（１２０に示す）クエリ実行層、（１３０に示す）トランザクション性・一貫性管理層、及び（１４０に示す）持続性・空き領域管理層を含む。上述したように、いくつかの既存のデータベースシステムにおいてデータベースインスタンスをスケーリングするには、データベースインスタンス全体（図１に示した層の全てを含む）を１回以上複製し、次いで、グルーロジックを加えてそれらを統合することが求められる場合がある。いくつかの実施形態では、本明細書で説明されたシステムは、その代わりに持続性・空き領域管理層１４０の機能をデータベース階層から別個のストレージ層に肩代わりさせてもよく、ストレージ層内の複数のストレージノードにわたってその機能を分散させてもよい。なお、開示されたトランザクションの順序付け技術は、持続性・空き領域管理層１４０がデータベース階層の一部である実施形態において適用されてもよい。

【００１７】

様々な実施形態では、本明細書で説明されたデータベースシステムは、様々なデータベースの動作／トランザクションのために、標準またはカスタムのアプリケーションプログラミングインターフェース（ａｐｐｌｉｃａｔｉｏｎ ｐｒｏｇｒａｍｍｉｎｇ ｉｎｔｅｒｆａｃｅ：ＡＰＩ）に対応し得る。例えば、このＡＰＩは、データベースを作成すること、テーブルを作成すること、テーブルを変更すること、ユーザーを作成すること、ユーザーを削除すること、テーブルに１つ以上の行を挿入すること、値をコピーすること、テーブル内からデータを選択すること（例えば、テーブルにクエリを行うこと）、クエリをキャンセルすること、またはクエリを中止することを行う操作、及び／または他の操作に対応してもよい。

【００１８】

いくつかの実施形態では、データベースインスタンスのデータベース階層は、読み取り要求及び／または書き込み要求（並びに／または他のトランザクション要求）を様々なクライアントプログラム（例えば、アプリケーション）及び／または加入者（ユーザー）から受け取り、次いで、それらを解析し、（単数または複数の）関連するデータベース動作を実行するための実行計画を作成するデータベースエンジン・ヘッドノードサーバ（プライマリノードと呼ぶこともある）を含んでもよい。例えば、このデータベースエンジン・ヘッドノードは、複合的なクエリ及びジョインに対する結果を得るのに必要な一連のステップを作成してもよい。いくつかの実施形態では、データベースエンジン・ヘッドノードは、データベース階層と別個の分散データベース最適化ストレージシステムとの間の通信に加えて、データベースシステムのデータベース階層とクライアント／加入者との間の通信を管理してもよい。いくつかの実施形態では、以下でより詳細に説明するように、データベースエンジン・ヘッドノードは、トランザクションの順序付けを行うように構成されてもよい。これにより、特定の分離レベル（例えば、読み取り一貫性など）を保持しやすくすることができる。

【００１９】
いくつかの実施形態では、データベース階層（またはより具体的には、データベースエンジン・ヘッドノード）は、最近アクセスされたデータページを一時的に保持するキャッシュを含んでもよい。このような実施形態では、かかるキャッシュ内のデータページを対象とする書き込み要求を受け取った場合、対応するＲＥＤＯログレコードをストレージ層に送り出すのに加えて、データベースエンジンは、自らのキャッシュ内のコピーにその変更を適用してもよい。しかしながら、他のデータベースシステムとは異なり、このキャッシュに保持されているデータページを、ストレージ層に絶えずフラッシュさせなくてもよく、このデータページは、任意の時点で（例えば、キャッシュ済みコピーに対して直近に適用された、書き込み要求のＲＥＤＯログレコードがストレージ層に送られて承認された後の任意の時点で）破棄され得る。このキャッシュは、種々の実施形態では、多くとも１つのライター（または複数のリーダー）によるキャッシュへの同時アクセスを制御するための様々なロッキング機構のいずれを実装してもよい。しかしながら、かかるキャッシュを含む実施形態では、複数のノードにわたってキャッシュを分散させなくてもよく、所与のデータベースインスタンスに対するデータベースエンジン・ヘッドノードにのみキャッシュが存在し得ることに留意されたい。従って、キャッシュのコヒーレンシまたは一貫性の問題を扱わなくてもよい。また、そうは言っても、それぞれがデータベースエンジン・ヘッドノードを備えた複数のデータベースインスタンスが存在してもよいことに留意されたい。

【００２０】
いくつかの実施形態では、データベース階層は、システム内の読み取り用レプリカ（例えば、読み取り要求がルーティングされる可能性のある、データベース階層の種々のノード上のデータの読み取り専用コピー）を同期または非同期で利用することに対応してもよい。このような実施形態では、所与のデータベーステーブルに対するデータベースエンジン・ヘッドノードは、特定のデータページを対象とする読み取り要求を受け取った場合、その要求を、これらの読み取り専用コピーのうちの任意の１つ（または特定の１つ）にルーティングしてもよい。いくつかの実施形態では、データベースエンジン・ヘッドノード内のクライアント側ドライバは、これらの他のノードに、（例えば、これらのノードに内部キャッシュの無効化を促し、その後、これらのノードが、更新済みデータページの更新済みコピーをストレージ層から要求できるようにするために）キャッシュ済みのデータページに対する更新及び／または無効化について通知するように構成されてもよい。

【００２１】
いくつかの実施形態では、クライアント側ドライバは、ボリュームに関するメタデータを保持してもよく、ストレージノード間にホップを追加せずとも、読み取り要求と書き込み要求を実行するのに必要なストレージノードのそれぞれに非同期要求を直接送ってもよい。例えば、いくつかの実施形態では、データベーステーブルを変更する要求に応答して、クライアント側ドライバは、対象データページ用のストレージを実装している１つ以上のノードを決定し、上記変更を指定する（単数または複数の）ＲＥＤＯログレコードをそれらのストレージノードにルーティングするように構成されてもよい。次いで、ストレージノードは、ＲＥＤＯログレコード内で指定された変更を、将来のある時点で対象データページに適用する役割を果たしてもよい。書き込みが承認されてクライアント側ドライバに返されると、クライアント側ドライバは、ボリュームが持続的である時点を進めて、データベース階層に返すコミットを承認してもよい。先に述べたように、いくつかの実施形態では、クライアント側ドライバは、データページをストレージノードサーバに絶えず送らなくてもよい。これにより、ネットワークトラフィックが減少し得るのみならず、前述のデータベースシステムでのフォアグラウンド処理能力を制約するチェックポイント・スレッドまたはバックグラウンド・ライタースレッドが不要ともなり得る。

【００２２】
いくつかの実施形態では、クライアント側ドライバは、本明細書で説明されるように、読み取り要求を受け取って複数のレコードを検索するデータベースエンジン・ヘッドノー

ドのために、開示されたトランザクションの順序付けを実行してもよい。例えば、データベースサービスのデータベースエンジン・ヘッドノードは、読み取り要求を受け取って、データベースサービスによって記憶されたレコードの読み取りを実行してもよい。別のデータベースエンジン・ヘッドノードは、トランザクション要求を受け取って、そのレコードに対してトランザクション（例えば、書き込みなど）を実行してもよい。読み取り要求を受け取ったデータベースエンジン・ヘッドノードは、その読み取りに関連付けられた時刻指示が、上記トランザクションに関連付けられた第２の時刻指示の閾値内にあると判定したことに基づき、潜在的な読み取り異常（例えば、ファジーリード、リードスキューなど）を検出し得る。潜在的な読み取り異常を検出したことに応答して、第１の時刻指示が第２の時刻指示よりも早い時点を示しているかどうかに関わらず、トランザクション要求によって指定されたトランザクションの実行後に、上記読み取りが実行され得る。ある事例では、読み取りを再試行することにより、その再試行については潜在的な読み取り異常が発生しないようにしてもよい。なお、データベースエンジン・ヘッドノードは、ある時刻に読み取り要求を受け取ってデータテーブルにクエリを行い、別の時刻にトランザクション要求を受け取ってデータテーブルを変更してもよい。以下で説明するように、通常リード、ファジーリード及びリードスキューに関する様々な例示的なタイミング図を図７Ａ～Ｃに示す。

【００２３】

いくつかの実施形態では、データベースエンジン・ヘッドノードのキャッシュによって多くの読み取り要求が提供されてもよい。しかしながら、書き込み要求には持続性が必要となる場合がある。これは、大規模な障害イベントがあまりにもよく起こるため、インメモリレプリケーションのみでは対処することができない場合があるためである。従って、本明細書で説明されたシステムは、ストレージ階層内のデータストレージを２つの領域として実装することにより、フォアグラウンド・レイテンシーパスに含まれるＲＥＤＯログレコードの書き込み動作のコストを最小化するように構成され得る。この２つの領域とは、すなわち、データベース階層からＲＥＤＯログレコードを受け取ったときにそれらが書き込まれる小さな追記専用のログ構造化領域、及びバックグラウンドにおいてデータページの新たなバージョンを作成するためにログレコードを１つに併合させたより大きな領域である。いくつかの実施形態では、インスタンス化されたデータブロックが参照されるまで、当該ページの後ろ向き連鎖のログレコードにおける最後のＲＥＤＯログレコードを指している各データページを対象としてインメモリ構造を維持してもよい。この手法により、読み取りが主にキャッシュされたアプリケーションにおけるものを含む、読み取り・書き込みが混在した作業負荷に対して良好な性能を得ることができる。

【００２４】

Ｗｅｂサービスベースのデータベースサービスを実装するように構成可能なサービスシステムアーキテクチャの一実施形態を図２に示す。例示した実施形態では、（データベースクライアント２５０ａ～２５０ｎと示した）いくつかのクライアントは、ネットワーク２６０を介してＷｅｂサービスプラットフォーム２００と相互作用するように構成され得る。Ｗｅｂサービスプラットフォーム２００は、データベースサービス２１０の１つ以上のインスタンス、分散データベース最適化ストレージサービス２２０及び／または１つ以上の他の仮想コンピューティングサービス２３０と相互作用するように構成され得る。なお、所与の構成要素の１つ以上のインスタンスが存在し得る場合、本明細書においてその構成要素への言及は、単数形または複数形のいずれでもなされることがある。しかしながら、いずれかの形式を用いることは、他の形式を排除することを意図するものではない。

【００２５】

様々な実施形態では、図２に示した構成要素は、コンピュータハードウェア内に直接実装され、コンピュータハードウェア（例えば、マイクロプロセッサまたはコンピュータシステム）によって直接的もしくは間接的に実行可能な命令として実装され、またはこれらの技術の組み合わせを用いて実装され得る。例えば、図２の構成要素は、いくつかのコンピューティングノード（または単に、ノード）を含むシステムによって実装され得る。こ

れらのノードのそれぞれは、図８に示され、以下で説明されるコンピュータシステムの実施形態と同様のものであってよい。様々な実施形態では、所与のサービスシステム構成要素（例えば、データベースサービスの構成要素またはストレージサービスの構成要素）の機能は、特定のノードによって実装されてもよく、またはいくつかのノードにわたって分散されてもよい。いくつかの実施形態では、所与のノードは、２つ以上のサービスシステム構成要素（例えば、２つ以上のデータベースサービスシステム構成要素）の機能を実装してもよい。

【００２６】

一般的に言えば、クライアント２５０は、データベースサービスを求める要求（例えば、トランザクション要求、読み取り要求など）を含むＷｅｂサービス要求を、Ｗｅｂサービスプラットフォーム２００にネットワーク２６０を介して送信するように構成可能な任意の種類のクライアントを含んでもよい。例えば、所与のクライアント２５０は、適切なバージョンのＷｅｂブラウザを搭載してもよく、または、Ｗｅｂブラウザによって提供される実行環境に対する拡張機能、もしくはその環境内の拡張機能として実行するように構成されたプラグインモジュールもしくは他の種類のコードモジュールを搭載してもよい。あるいは、クライアント２５０（例えば、データベースサービスクライアント）は、永続的記憶装置のリソースを利用して１つ以上のデータベーステーブルを記憶し、かつ／またはそれらにアクセスし得る、データベースアプリケーションなどのアプリケーション（またはそのユーザーインターフェース）、メディアアプリケーション、オフィスアプリケーション、またはその他のアプリケーションを含んでもよい。いくつかの実施形態では、こうしたアプリケーションは、（例えば、適切なバージョンのＨＴＴＰ（Ｈｙｐｅｒｔｅｘｔ　Ｔｒａｎｓｆｅｒ　Ｐｒｏｔｏｃｏｌ）に対して）十分なプロトコルサポートを含むことにより、全ての種類のＷｅｂベースのデータ用に完全なブラウザサポートを必ずしも実装せずに、Ｗｅｂサービス要求を生成及び処理できるようにしてもよい。すなわち、クライアント２５０は、Ｗｅｂサービスプラットフォーム２００と直接相互作用するように構成されたアプリケーションであってよい。いくつかの実施形態では、クライアント２５０は、ＲＥＳＴ（Ｒｅｐｒｅｓｅｎｔａｔｉｏｎａｌ　Ｓｔａｔｅ　Ｔｒａｎｓｆｅｒ）スタイルのＷｅｂサービスアーキテクチャ、ドキュメントベースのもしくはメッセージベースのＷｅｂサービスアーキテクチャ、または他の適切なＷｅｂサービスアーキテクチャに従って、Ｗｅｂサービス要求を生成するように構成されてもよい。

【００２７】

いくつかの実施形態では、クライアント２５０（例えば、データベースサービスクライアント）は、上記アプリケーションに透過的な方法で、データベーステーブルの、Ｗｅｂサービスベースのストレージに対するアクセスを他のアプリケーションに提供するように構成されてもよい。例えば、クライアント２５０は、本明細書で説明されたストレージモデルの適切な変形例に従ってストレージを提供するオペレーティングシステムまたはファイルシステムと一体化されるように構成されてもよい。しかしながら、このオペレーティングシステムまたはファイルシステムは、ファイル、ディレクトリ及び／またはフォルダで構成される従来のファイルシステム階層構造などの、別のストレージインターフェースをアプリケーションに提示してもよい。こうした実施形態では、図１のストレージシステムサービスモデルを利用するためにアプリケーションを修正しなくてもよい。むしろ、Ｗｅｂサービスプラットフォーム２００へのインターフェースの詳細については、オペレーティングシステム環境内で実行するアプリケーションに代わり、クライアント２５０及びオペレーティングシステムまたはファイルシステムによって調整され得る。

【００２８】

クライアント２５０は、ネットワーク２６０を介し、Ｗｅｂサービス要求（例えば、トランザクション要求、読み取り要求など）をＷｅｂサービスプラットフォーム２００に伝達してもよく、そこから応答を受け取ってもよい。様々な実施形態では、ネットワーク２６０は、クライアント２５０とプラットフォーム２００の間のＷｅｂベース通信を確立するのに必要なネットワーク接続ハードウェアとネットワーク接続プロトコルの任意の適切

な組み合わせを含んでもよい。例えば、ネットワーク２６０は、インターネットを集合的に実装する様々な遠距離通信ネットワーク及びサービスプロバイダを通常含んでもよい。ネットワーク２６０は、パブリックまたはプライベートな無線ネットワークに加えて、ローカルエリアネットワーク（ｌｏｃａｌ　ａｒｅａ　ｎｅｔｗｏｒｋ：ＬＡＮ）またはワイドエリアネットワーク（ｗｉｄｅ　ａｒｅａ　ｎｅｔｗｏｒｋ：ＷＡＮ）などのプライベートネットワークを含んでもよい。例えば、所与のクライアント２５０とＷｅｂサービスプラットフォーム２００の両方を、内部ネットワークを独自に有する企業内にそれぞれ提供してもよい。こうした実施形態では、ネットワーク２６０は、インターネットとＷｅｂサービスプラットフォーム２００の間に加えて、所与のクライアント２５０とインターネットの間でネットワーク接続リンクを確立するのに必要な、ハードウェア（例えば、モデム、ルーター、スイッチ、負荷分散装置、プロキシサーバなど）、及びソフトウェア（例えば、プロトコルスタック、会計ソフトウェア、ファイアーウォール／セキュリティソフトウェアなど）を含んでもよい。なお、いくつかの実施形態では、クライアント２５０は、パブリックインターネットではなくプライベートネットワークを用いてＷｅｂサービスプラットフォーム２００と通信してもよい。例えば、クライアント２５０は、データベースサービスシステム（例えば、データベースサービス２１０及び／または分散データベース最適化ストレージサービス２２０を実装するシステム）として同一企業内に提供されてもよい。その場合、クライアント２５０は、プライベートネットワーク２６０（例えば、インターネットベースの通信プロトコルを利用し得るが、公にアクセス可能ではないＬＡＮまたはＷＡＮ）全体を通じてプラットフォーム２００と通信してもよい。

【００２９】

一般的に言えば、Ｗｅｂサービスプラットフォーム２００は、データページ（またはそのレコード）にアクセスする要求などのＷｅｂサービス要求を受け取って処理するように構成された１つ以上のサービスエンドポイントを実装するように構成され得る。例えば、Ｗｅｂサービスプラットフォーム２００は、特定のエンドポイントを実装するように構成されたハードウェア及び／またはソフトウェアを含むことにより、そのエンドポイントを対象とするＨＴＴＰベースのＷｅｂサービス要求を正しく受け取って処理できるようにしてもよい。一実施形態では、Ｗｅｂサービスプラットフォーム２００は、クライアント２５０からＷｅｂサービス要求を受け取り、それらを、データベースサービス２１０、分散データベース最適化ストレージサービス２２０及び／または別の仮想コンピューティングサービス２３０を実装するシステムの構成要素に転送して処理できるように構成されたサーバシステムとして実装されてもよい。他の実施形態では、Ｗｅｂサービスプラットフォーム２００は、大量のＷｅｂサービス要求処理負荷を動的に管理するように構成された負荷分散機能及び他の要求管理機能を実装するいくつかの別個のシステムとして（例えば、クラスタートポロジーを用いて）構成されてもよい。様々な実施形態では、Ｗｅｂサービスプラットフォーム２００は、ＲＥＳＴスタイルまたはドキュメントベースの（例えば、ＳＯＡＰベースの）種類のＷｅｂサービス要求に対応するように構成されてもよい。

【００３０】

クライアントのＷｅｂサービス要求に対してアドレス可能エンドポイントとして機能することに加えて、いくつかの実施形態では、Ｗｅｂサービスプラットフォーム２００は、様々なクライアント管理機能を実装してもよい。例えば、プラットフォーム２００は、要求中のクライアント２５０の識別情報、クライアント要求の数及び／または頻度、クライアント２５０に代わって記憶もしくは検索されたデータテーブル（もしくはそのレコード）のサイズ、クライアント２５０によって使用されるストレージ帯域全体、クライアント２５０によって要求されたストレージのクラス、またはその他の測定可能なクライアントの使用状況パラメータを追跡することなどにより、ストレージリソースを含めた、クライアントによるＷｅｂサービスの使用状況の測定と計算とを連携して行うようにしてもよい。プラットフォーム２００は、いくつかの実施形態では、クライアントのＷｅｂサービス要求を、このプラットフォームの各データベースインスタンスの特定のデータベースエンジン・ヘッドノードに分配するように構成されてもよい。簡単な例として、時刻１にて、

プラットフォーム２００は、読み取り要求をデータベースエンジン・ヘッドノード１に分配してもよく、時刻３にて、プラットフォームは、書き込み要求をデータベースエンジン・ヘッドノード２に分配してもよい。プラットフォーム２００は、財務会計システム及び課金システムを実装してもよく、または使用状況データのデータベースを保持してもよい。この使用状況データは、クライアントの利用活動に対する報告及び課金を行う外部システムによって照会及び処理が行われる。ある実施形態では、プラットフォーム２００は、様々なストレージサービスシステムの運用上の測定基準を収集、監視、及び／または集約するように構成されてもよい。このような測定基準には、クライアント２５０から受け取った要求の割合及び種類を反映している測定基準、かかる要求によって利用される帯域、かかる要求に対するシステムの処理レイテンシー、システム構成要素の利用率（例えば、ストレージサービスシステム内のネットワーク帯域及び／もしくはストレージの利用率）、要求に起因するエラーの割合及び種類、記憶され、要求されたデータページもしくはそのレコードの特性（例えば、サイズ、データ種類など）、またはその他の適切な測定基準などがある。いくつかの実施形態では、こうした測定基準は、システム構成要素の調整及び保守を行うためにシステム管理者によって用いられてもよい。一方、他の実施形態では、こうした測定基準（または、当該測定基準の関連部分）をクライアント２５０に公開することにより、当該クライアントが、データベースサービス２１０、分散データベース最適化ストレージサービス２２０及び／または別の仮想コンピューティングサービス２３０（またはそれらのサービスを実装する下位システム）における自らの使用状況を監視できるようにしてもよい。

【００３１】

いくつかの実施形態では、プラットフォーム２００は、ユーザー認証手順及びユーザーアクセス制御手順を実装してもよい。例えば、特定のデータベーステーブルにアクセスする所与のＷｅｂサービス要求に対して、プラットフォーム２００は、その要求に関連付けられたクライアント２５０が当該特定のデータベーステーブルにアクセスする権限を与えられているかどうかを確認するように構成され得る。プラットフォーム２００は、こうした認証を、例えば、当該特定のデータベーステーブルに関連付けられた証明書と照合して識別情報、パスワードもしくは他の証明書を評価すること、または当該特定のデータベーステーブル用のアクセス制御リストと照合して、要求された当該特定のデータベーステーブルへのアクセスを評価することによって判定してもよい。例えば、当該特定のデータベーステーブルにアクセスするための正当な証明書をクライアント２５０が持っていない場合、プラットフォーム２００は、例えば、要求中のクライアント２５０に対してエラー状態を示す応答を返すことにより、対応するＷｅｂサービス要求を拒絶してもよい。様々なアクセス制御ポリシーが、アクセス制御情報のレコードまたはリストとして、データベースサービス２１０、分散データベース最適化ストレージサービス２２０及び／または仮想コンピューティングサービス２３０によって記憶されてもよい。

【００３２】

なお、Ｗｅｂサービスプラットフォーム２００は、データベースサービス２１０を実装するデータベースシステムの機能にクライアント２５０がアクセスできるようにするための主要なインターフェースに相当し得るが、かかる機能への唯一のインターフェースに相当する必要はない。例えば、Ｗｅｂサービスインターフェースとは異なる代替ＡＰＩを用いることにより、データベースシステムを提供する企業内部のクライアントが、Ｗｅｂサービスプラットフォーム２００を回避できるようにしてもよい。なお、本明細書で説明された実施例の多くでは、分散データベース最適化ストレージサービス２２０は、データベースサービスをクライアント２５０に提供するコンピューティングシステムまたは企業システムの内部にあってよく、外部のクライアント（例えば、ユーザーまたはクライアントアプリケーション）に公開されなくてもよい。このような実施形態では、内部の「クライアント」（例えば、データベースサービス２１０）は、分散データベース最適化ストレージサービス２２０とデータベースサービス２１０の間の実線で示すように、ローカルネットワークまたはプライベートネットワークを介して分散データベース最適化ストレージサ

ービス２２０に（例えば、ＡＰＩを通じて、これらのサービスを実装するシステム間で直接的に）アクセスしてもよい。このような実施形態では、データベーステーブルに記憶する際にクライアント２５０に代わって分散データベース最適化ストレージサービス２２０を利用することは、それらのクライアントにとって透過的に行われてもよい。他の実施形態では、Ｗｅｂサービスプラットフォーム２００を通じてクライアント２５０に分散データベース最適化ストレージサービス２２０を公開することにより、データベースを管理するためのデータベースサービス２１０に依存するアプリケーション以外のアプリケーションに対し、データベーステーブルまたは他の情報のストレージを提供してもよい。これは、図２においてＷｅｂサービスプラットフォーム２００と分散データベース最適化ストレージサービス２２０の間の点線によって示されている。このような実施形態では、分散データベース最適化ストレージサービス２２０のクライアントは、分散データベース最適化ストレージサービス２２０にネットワーク２６０経由で（例えば、インターネットを介して）アクセスしてもよい。いくつかの実施形態では、仮想コンピューティングサービス２３０は、コンピューティングサービス２３０を実行するのに用いられるオブジェクトをクライアント２５０に代わって記憶する分散データベース最適化ストレージサービス２２０からストレージサービスを（例えば、ＡＰＩを通じて、仮想コンピューティングサービス２３０と分散データベース最適化ストレージサービス２２０の間で直接的に）受けるように構成され得る。これは、図２において仮想コンピューティングサービス２３０と分散データベース最適化ストレージサービス２２０の間の点線によって示されている。ある場合には、プラットフォーム２００の会計サービス及び／または証明書発行サービスは、管理用クライアントなどの内部のクライアントにとっては、または同一企業内のサービス構成要素間では不要となる場合がある。

【００３３】

なお、様々な実施形態では、種々のストレージポリシーが、データベースサービス２１０及び／または分散データベース最適化ストレージサービス２２０によって実装され得る。このようなストレージポリシーの実施例としては、持続性ポリシー（例えば、記憶されるデータベーステーブル（もしくはそのデータページ）のインスタンスの数、及びそれらが記憶される種々のノードの数を示すポリシー）、並びに／または負荷分散ポリシー（このポリシーにより、データベーステーブルもしくはそのデータページが、要求のトラフィックを一様にするために種々のノード、ボリューム及び／またはディスクにわたって分散され得る）を挙げることができる。加えて、上記サービスの中の多様なサービスにより、各種の記憶済み項目に異なるストレージポリシーを適用してもよい。例えば、いくつかの実施形態では、分散データベース最適化ストレージサービス２２０は、データページの持続性よりもＲＥＤＯログレコードの持続性が高くなるように実装されてもよい。

【００３４】

図３は、一実施形態に係る、データベースエンジン及び別個の分散データベースストレージサービスを含むデータベースシステムの様々な構成要素を示すブロック図である。本実施例では、データベースシステム３００は、いくつかのデータベーステーブルのそれぞれに対する各データベースエンジン・ヘッドノード３２０、及び分散データベース最適化ストレージサービス３１０（これは、データベースクライアント３５０ａ～３５０ｎとして示される、データベースシステムのクライアントにとって見えても見えなくてもよい）を含む。本実施例に示すように、データベースクライアント３５０ａ～３５０ｎの１つ以上が、データベースヘッドノード３２０（例えば、ヘッドノード３２０ａ、ヘッドノード３２０ｂまたはヘッドノード３２０ｃであって、これらのノードのそれぞれは、各データベースインスタンスの構成要素である）にネットワーク３６０経由でアクセスしてもよい（例えば、これらの構成要素は、ネットワークアドレス可能かつデータベースクライアント３５０ａ～３５０ｎにとって利用可能であってよい）。しかしながら、異なる実施形態では、分散データベース最適化ストレージサービス３１０は、データベースシステムによって利用されて、１つ以上のデータベーステーブルのデータページ（並びにＲＥＤＯログレコード及び／またはそれに関連付けられた他のメタデータ）をデータベースクライアン

ト３５０ａ～３５０ｎに代わって記憶し、本明細書で説明されるデータベースシステムの他の機能を実行し得るが、ネットワークアドレス可能かつストレージクライアント３５０ａ～３５０ｎにとって利用可能であってもよく、そうでなくてもよい。例えば、いくつかの実施形態では、分散データベース最適化ストレージサービス３１０は、ストレージクライアント３５０ａ～３５０ｎにとって見えないようにして、様々な記憶、アクセス、変更ロギング、リカバリ、ログレコード操作、及び／または空き領域管理動作を行ってもよい。

【００３５】

先に述べたように、各データベースインスタンスは、１つのデータベースエンジン・ヘッドノード３２０を含み得る。このヘッドノードは、要求（例えば、トランザクション要求など）を様々なクライアントプログラム（例えば、アプリケーション）及び／または加入者（ユーザー）から受け取り、次いでそれらを解析し、それらを最適化し、実行計画に展開することにより、関連するデータベース動作（単数または複数）を実行する。図３に示した例では、データベースエンジン・ヘッドノード３２０ａの、クエリの解析・最適化・実行用構成要素３０５は、データベースクライアント３５０ａから受け取ったクエリであって、データベースエンジン・ヘッドノード３２０ａが構成要素になっているデータベースインスタンスを対象とするクエリに対してこれらの機能を実行し得る。いくつかの実施形態では、クエリの解析・最適化・実行用構成要素３０５は、データベースクライアント３５０ａにクエリ応答を返してもよい。この応答には、必要に応じて、書き込み承認、要求されたデータページ（もしくはその各部）、エラーメッセージ及びまたは他の応答が含まれ得る。本実施例に示すように、データベースエンジン・ヘッドノード３２０ａは、クライアント側ストレージサービスドライバ３２５を含んでもよい。このストレージサービスドライバは、読み取り要求及び／または（例えば、書き込みの）ＲＥＤＯログレコードを分散データベース最適化ストレージサービス３１０内の様々なストレージノードにルーティングし、書き込み承認を分散データベース最適化ストレージサービス３１０から受け取り、要求されたデータページを分散データベース最適化ストレージサービス３１０から受け取り、かつ／またはデータページ、エラーメッセージもしくは他の応答をクエリの解析・最適化・実行用構成要素３０５に返してもよい（さらに、この構成要素は、このデータページ、エラーメッセージまたは他の応答をデータベースクライアント３５０ａに返してもよい）。

【００３６】

本実施例では、データベースエンジン・ヘッドノード３２０ａはデータページキャッシュ３３５を含んでおり、その内部には、最近アクセスされたデータページが一時的に保持され得る。図３に示すように、データベースエンジン・ヘッドノード３２０ａは、トランザクション・一貫性管理構成要素３３０を含んでもよい。この構成要素は、データベースエンジン・ヘッドノード３２０ａが構成要素になっているデータベースインスタンスにおいてトランザクション性及び一貫性を提供する役割を果たしてもよい。例えば、この構成要素は、データベースインスタンスの原子性、一貫性及び独立性の諸特性、並びにそのデータベースインスタンス対象とするトランザクションを確保する役割を果たしてもよい。例えば、データベースサービスのデータベースエンジン・ヘッドノードは、読み取り要求を受け取って、データベースサービスによって記憶されたレコードの読み取りを実行してもよい。別のデータベースエンジン・ヘッドノードは、トランザクション要求を受け取って、そのレコードに対してトランザクション（例えば、書き込みなど）を実行してもよい。次いで、読み取り要求を受け取ったデータベースエンジン・ヘッドノードのトランザクション・一貫性管理構成要素３３０は、その読み取りに関連付けられた時刻指示が、上記トランザクションに関連付けられた第２の時刻指示の閾値内にあると判定したことに基づき、潜在的な読み取り異常（例えば、ファジーリード、リードスキューなど）を検出し得る。潜在的な読み取り異常を検出したことに応答して、第１の時刻指示が第２の時刻指示よりも早い時点を示しているかどうかに関わらず、トランザクション要求によって指定されたトランザクションの後に上記読み取りが実行され得る。場合によっては、読み取りを

再試行することにより、その再試行については潜在的な読み取り異常が発生しないようにしてもよい。

【００３７】

図３に示すように、データベースエンジン・ヘッドノード３２０ａは、トランザクションログ３４０及びＵＮＤＯログ３４５を含んでもよい。これらのログをトランザクション・一貫性管理構成要素３３０によって利用することにより、様々なトランザクションのステータスを追跡し、コミットされていないトランザクションに関する、ローカルにキャッシュされた任意の結果をロールバックしてもよい。

【００３８】

なお、図３に示した他のデータベースエンジン・ヘッドノード３２０のそれぞれ（例えば、３２０ｂ及び３２０ｃ）は同様の構成要素を含んでもよく、当該データベースエンジン・ヘッドノードが構成要素になっている各データベースインスタンスを対象とし、かつデータベースクライアント３５０ａ～３５０ｎの１つ以上によって受け取られるクエリ及び／または他のトランザクションに対して同様の機能を実行してもよい。例えば、開示されたトランザクションの順序付け技術は、本明細書で説明するように、２つの異なるデータベースエンジン・ヘッドノードが、閾時間内に同一データにアクセスしている（例えば、一方が読み取り中であり、一方が書き込み中である）シナリオにおいて実施され得る。

【００３９】

分散データベース最適化ストレージシステムの一実施形態をブロック図によって図４に示す。本実施例では、データベースシステム４００は、分散データベース最適化ストレージシステム４１０を含んでおり、このストレージシステムは、インターコネクト４６０を介してデータベースエンジン・ヘッドノード４２０と通信する。図３に示した実施例と同様に、データベースエンジン・ヘッドノード４２０は、クライアント側ストレージサービスドライバ４２５を含んでもよい。本実施例では、分散データベース最適化ストレージシステム４１０は、（４３０、４４０及び４５０として示したものを含む）複数のストレージシステムサーバノードを含む。これらのノードのそれぞれは、各ノードが記憶する（単数または複数の）セグメントに対応したＲＥＤＯログ及びデータページ用のストレージ、並びに様々なセグメント管理機能を実行するように構成されたハードウェア及び／またはソフトウェアを含む。例えば、各ストレージシステムサーバノードは、以下の動作：（ローカル的に、例えば、ストレージノード内で行われる）複製、データページを生成するためのＲＥＤＯログの併合、ログ管理（例えば、ログレコードの操作）、クラッシュリカバリ、及び／または（例えば、セグメントに対する）空き領域管理、のいずれかまたは全ての少なくとも一部を実行するように構成されたハードウェア及び／またはソフトウェアを含んでもよい。各ストレージシステムサーバノードは、複数の付属ストレージ装置（例えば、ＳＳＤ）を備えてもよい。このストレージ装置には、クライアント（例えば、ユーザー、クライアントアプリケーション、及び／またはデータベースサービスの加入者）に代わってデータブロックが記憶されてもよい。

【００４０】

図４に示した実施例では、ストレージシステムサーバノード４３０は、（単数または複数の）データページ４３３、（単数または複数の）セグメントＲＥＤＯログ４３５、セグメント管理機能４３７及び付属ＳＳＤ４７１～４７８を含む。さらになお、「ＳＳＤ」という表示は、その下位ハードウェアに関係なくソリッドステートドライブを表しても表さなくてもよいが、より一般的にはローカルのブロックストレージボリュームを表し得る。同様に、ストレージシステムサーバノード４４０は、データページ（単数または複数）４４３、（単数または複数の）セグメントＲＥＤＯログ４４５、セグメント管理機能４４７及び付属ＳＳＤ４８１～４８８を含み、ストレージシステムサーバノード４５０は、（単数または複数の）データページ４５３、（単数または複数の）セグメントＲＥＤＯログ４５５、セグメント管理機能４５７及び付属ＳＳＤ４９１～４９８を含む。

【００４１】

いくつかの実施形態では、分散データベース最適化ストレージシステム内のストレージ

システムサーバノードのそれぞれは、一式のプロセスを実装し得る。これらのプロセスは、ノードサーバのオペレーティングシステム上で動作するものであり、このオペレーティングシステムは、データベースエンジン・ヘッドノードとの通信を管理して、例えば、ＲＥＤＯログを受け取り、データページを送り返すことなどを行う。いくつかの実施形態では、分散データベース最適化ストレージシステムに書き込まれた全データブロックを、（例えば、キー値を利用した永続的なリモートバックアップストレージシステムを用いて）長期ストレージ及び／またはアーカイブストレージにバックアップしてもよい。

【００４２】

図５は、一実施形態に係る、データベースシステム内の別個の分散データベース最適化ストレージシステムの利用を示すブロック図である。本実施例では、１つ以上のクライアントプロセス５１０は、データベースエンジン５２０及び分散データベース最適化ストレージシステム５３０を含むデータベースシステムによって保持される１つ以上のデータベーステーブルにデータを記憶し得る。図５に示した実施例では、データベースエンジン５２０は、データベース階層要素５６０、及びクライアント側ドライバ５４０（このドライバは、分散データベース最適化ストレージシステム５３０とデータベース階層要素５６０の間のインターフェースとして機能する）を含む。いくつかの実施形態では、データベース階層要素５６０は、図３のクエリの解析・最適化・実行用構成要素３０５及びトランザクション・一貫性管理構成要素３３０（例えば、トランザクションの順序付け）によって実行されるものなどの機能を実行してもよく、かつ／またはデータページ、トランザクションログ及び／もしくはＵＮＤＯログ（図３のデータページキャッシュ３３５、トランザクションログ３４０及びＵＮＤＯログ３４５によって記憶されるものなど）を記憶してもよい。

【００４３】

本実施例では、１つ以上のクライアントプロセス５１０は、データベース・クエリ要求５１５（この要求は、ストレージノード５３５ａ～５３５ｎの１つ以上に記憶されたデータを対象とする、読み取り要求、及び／または書き込み要求、及び／または他のトランザクション要求、を含み得る）をデータベース階層要素５６０に送ってもよく、（例えば、書き込み承認及び／または要求されたデータを含む）データベース・クエリ応答５１７をデータベース階層要素５６０から受け取ってもよい。データページに書き込む要求を含む各データベース・クエリ要求５１５を解析及び最適化して、１つ以上の書き込みレコード要求５４１が生成され得る。これらの要求は、引き続いて分散データベース最適化ストレージシステム５３０にルーティングできるように、クライアント側ドライバ５４０に送られてもよい。本実施例では、クライアント側ドライバ５４０は、各書き込みレコード要求５４１に対応する１つ以上のＲＥＤＯログレコード５３１を生成してもよく、それらを、分散データベース最適化ストレージシステム５３０を構成するストレージノード５３５の特定のストレージノードに送ってもよい。分散データベース最適化ストレージシステム５３０は、各ＲＥＤＯログレコード５３１について、対応する書き込み承認５３２をデータベースエンジン５２０に（具体的には、クライアント側ドライバ５４０に）返してもよい。クライアント側ドライバ５４０は、これらの書き込み承認をデータベース階層要素５６０に（書き込み応答５４２として）渡してもよく、次いで、このデータベース階層要素は、対応する応答（例えば、書き込み承認）を１つ以上のクライアントプロセス５１０にデータベース・クエリ応答５１７の１つとして送ってもよい。

【００４４】

本実施例では、データページを読み取る要求を含む各データベース・クエリ要求５１５を解析及び最適化して、１つ以上の読み取りレコード要求５４３が生成され得る。これらの要求は、引き続いて分散データベース最適化ストレージシステム５３０にルーティングできるように、クライアント側ドライバ５４０に送られてもよい。本実施例では、クライアント側ドライバ５４０は、これらの要求を、分散データベース最適化ストレージシステム５３０を構成するストレージノード５３５の特定のストレージノードに送ってもよく、分散データベース最適化ストレージシステム５３０は、要求されたデータページ５３３を

データベースエンジン５２０に（具体的には、クライアント側ドライバ５４０に）返してもよい。クライアント側ドライバ５４０は、返されたデータページをデータベース階層要素５６０に返却データレコード５４４として送ってもよく、次いで、データベース階層要素５６０は、そのデータページを１つ以上のクライアントプロセス５１０にデータベース・クエリ応答５１７として送ってもよい。

【００４５】

いくつかの実施形態では、様々なエラー及び／またはデータ損失のメッセージ５３４を分散データベース最適化ストレージシステム５３０からデータベースエンジン５２０に（具体的には、クライアント側ドライバ５４０に）送ってもよい。これらのメッセージは、クライアント側ドライバ５４０からデータベース階層要素５６０にエラー及び／または損失の報告メッセージ５４５として渡されてもよく、次いで、データベース・クエリ応答５１７と共に（またはその代わりに）１つ以上のクライアントプロセス５１０に渡されてもよい。

【００４６】

いくつかの実施形態では、分散データベース最適化ストレージシステム５３０のＡＰＩ５３１～５３４、及びクライアント側ドライバ５４０のＡＰＩ５４１～５４５は、データベースエンジン５２０が分散データベース最適化ストレージシステム５３０のクライアントであるように、分散データベース最適化ストレージシステム５３０の機能をデータベースエンジン５２０に公開してもよい。例えば、（クライアント側ドライバ５４０を介した）データベースエンジン５２０は、これらのＡＰＩを通じてＲＥＤＯログレコードを書き込むこと、またはデータページを要求することにより、データベースエンジン５２０と分散データベース最適化ストレージシステム５３０の組み合わせによって実装されるデータベースシステムの様々な動作（例えば、記憶、アクセス、変更ロギング、リカバリ、及び／または空き領域管理動作）を実行する（またはその性能を促進する）ようにしてもよい。図５に示すように、分散データベース最適化ストレージシステム５３０は、ストレージノード５３５ａ～５３５ｎにデータブロックを記憶してもよく、これらのノードのそれぞれは、複数の付属ＳＳＤを備えてもよい。いくつかの実施形態では、分散データベース最適化ストレージシステム５３０は、各種の冗長方式を適用することにより、記憶されたデータブロックに対して高度な持続性を提供してもよい。

【００４７】

なお、様々な実施形態では、図５における、データベースエンジン５２０と分散データベース最適化ストレージシステム５３０の間のＡＰＩコール及びＡＰＩ応答（例えば、ＡＰＩ５３１～５３４）及び／またはクライアント側ドライバ５４０とデータベース階層要素５６０の間のＡＰＩコール及びＡＰＩ応答（例えば、ＡＰＩ５４１～５４５）は、安全なプロキシ接続（例えば、ゲートウェイ制御プレーンによって管理される接続）を介して実行されてもよく、または、パブリックネットワークを介して、もしくは代替的に、バーチャルプライベートネットワーク（ｖｉｒｔｕａｌ　ｐｒｉｖａｔｅ　ｎｅｔｗｏｒｋ：ＶＰＮ）接続などのプライベートチャネルを介して実行されてもよい。本明細書で説明されたデータベースシステムの構成要素に対する、かつ／またはこれらの構成要素間の上記の及び他のＡＰＩは、ＳＯＡＰ（Ｓｉｍｐｌｅ　Ｏｂｊｅｃｔ　Ａｃｃｅｓｓ　Ｐｒｏｔｏｃｏｌ）技術及びＲＥＳＴ（Ｒｅｐｒｅｓｅｎｔａｔｉｏｎａｌ　ｓｔａｔｅ　ｔｒａｎｓｆｅｒ）技術を含む種々の技術に従って実装されてもよいが、これらに限定されることはない。例えば、これらのＡＰＩは、ＳＯＡＰ　ＡＰＩまたはＲＥＳＴｆｕｌ　ＡＰＩとして実装されてもよいが、必ずしもそのように実装されるとは限らない。ＳＯＡＰは、Ｗｅｂベースのサービスに関連して情報を交換するためのプロトコルである。ＲＥＳＴは、分散ハイパーメディアシステム用の構築様式である。ＲＥＳＴｆｕｌ　ＡＰＩ（ＲＥＳＴｆｕｌ　Ｗｅｂサービスと呼ぶこともある）は、ＨＴＴＰ及びＲＥＳＴ技術を用いて実装されるＷｅｂサービスＡＰＩである。本明細書で説明されたＡＰＩは、いくつかの実施形態では、Ｃ、Ｃ＋＋、Ｊａｖａ、Ｃ＃及びＰｅｒｌを含むがこれらに限定されない様々な言語を用いたクライアントライブラリによってラップされて、データベースエンジン５

２０及び／または分散データベース最適化ストレージシステム５３０との一体化に対応し得る。

【００４８】

上述したように、いくつかの実施形態では、データベースシステムの機能要素を、データベースエンジンによって実行される機能要素と、別個の分散データベース最適化ストレージシステムにおいて実行される機能要素との間で分割してもよい。ある具体的な例では、あるものをデータベーステーブルに挿入するために（例えば、１つのデータブロックにレコードを追加することによってそのデータブロックを更新するために）クライアントプロセス（またはそのスレッド）から要求を受け取ったことに応答して、データベースエンジン・ヘッドノードの１つ以上の構成要素は、クエリの解析、最適化及び実行を行ってもよく、そのクエリの各部を、トランザクション・一貫性管理構成要素に送ってもよい。

【００４９】

トランザクション・一貫性管理構成要素は、同時に同一行を修正しようとしているクライアントプロセス（またはそのスレッド）が他にないことを保証し得る。例えば、トランザクション・一貫性管理構成要素は、この変更が原子的に、一貫して、持続的に、かつそのデータベース内で独立して実行されることを保証する役割を果たし得る。例えば、トランザクション・一貫性管理構成要素は、データベースエンジン・ヘッドノードのクライアント側ストレージサービスと連携して動作することにより、分散データベース最適化ストレージサービス内のノードの１つに送るべきＲＥＤＯログレコードを生成し、それを、このトランザクションに対してＡＣＩＤ特性が確実に満たされる順序及び／またはタイミングで、（他のクライアント要求に応答して生成された他のＲＥＤＯログと共に）分散データベース最適化ストレージサービスに送ってもよい。そのＲＥＤＯログレコードを受け取ると、対応するストレージノードは、該当するデータブロックを更新してもよく、そのデータブロックに対するＲＥＤＯログ（例えば、そのデータブロックを対象とする全ての変更が記録されたレコード）を更新してもよい。いくつかの実施形態では、データベースエンジンは、この変更に対するＵＮＤＯログレコードを生成する役割を果たしてもよく、そのＵＮＤＯログに対するＲＥＤＯログレコードを生成する役割もまた果たしてもよい。その上で、それらのレコードの両方をローカル的に（データベース階層内で）利用して、トランザクション性を保証するようにしてもよい。さらに、様々な実施形態では、トランザクション・一貫性管理構成要素は、トランザクションの順序付けを行うように構成され得る。例えば、トランザクション・一貫性管理構成要素は、ほぼ同時に（例えば、互いの閾値内の一貫性ポイントで）複数のデータベースエンジン・ヘッドノードがトランザクションの実行（例えば、読み取り及びコミット）を試みる状況において、潜在的な読み取り異常（例えば、ファジーリード、リードスキューなど）を検出するように構成され得る。潜在的な読み取り異常を検出したことに応答して、トランザクション・一貫性管理構成要素はさらに、その読み取りがより早い時刻に関連付けられている場合であっても、他のトランザクションの後に上記読み取りを発生させるように構成され得る。

【００５０】

次に、図６に目を向けると、様々な実施形態では、データベースシステム３００（または、データの読み取り、書き込み及び記憶に使用可能なデータベースサービス以外のあるシステム）は、トランザクションの順序付けを行うように構成されてもよい。図６の方法は、データベースエンジン・ヘッドノード３２０ａ、３２０ｂ、３２０ｃなどのトランザクション・一貫性管理構成要素３３０及び／またはクライアント側ドライバなどの、分散データベースシステムの様々な構成要素（例えばノード）によって行われるものとして説明され得るが、この方法は、ある場合には特定の構成要素によって行われる必要はない。例えば、ある場合には、図６の方法は、いくつかの実施形態に係る、他の構成要素またはコンピュータシステムによって行われてもよい。あるいは、場合によっては、データベースシステム３００の構成要素は、図３の実施例に示したものとは異なるように、組み合わされてもよく、または存在してもよい。様々な実施形態では、図６の方法は、分散データベースシステムの１つ以上のノードによって行われてもよく、これらのノードの１つが図

８のコンピュータシステムとして示されている。図６の方法は、トランザクションの順序付けを行う方法の例示的な一実施態様として示されている。他の実施態様では、図６の方法は、図示したブロック以外の追加ブロック、または図示したブロックよりも少ないブロックを含み得る。

【００５１】

６１０では、（例えば、データベースサービスまたは他のサービスによって記憶された）レコードの読み取りを実行する読み取り要求、及びそのレコードに対するトランザクションを実行するトランザクション要求を、例えば、（例えば、データベースサービスまたは他のサービスの）１つ以上のクライアントから受け取ってもよい。一実施形態では、この読み取り要求を、ＳＥＬＥＣＴステートメントまたは他の要求として受け取ってもよい。この読み取り要求は、レコードのスナップショットを見ることができるスナップショットポイントの時点を求める要求と呼ばれることもある。上記トランザクション要求は、ＵＰＤＡＴＥ、ＩＮＳＥＲＴ、または（例えば、データベースの）レコードを変更するのに使用可能な他のトランザクション（例えば、書き込みトランザクション）であってもよく、そのトランザクション要求をコミットしてもよい。様々な実施形態では、トランザクション要求及び読み取り要求は、記憶されているレコードに対して同時にアクセスし得る種々のノード（例えば、読み取り／書き込みを行うことができる複数のプライマリノード及び／または１つのプライマリノード、並びにその時点でレコードの読み取りのみが可能な読み取り用レプリカ）によって受け取られてもよい。例えば、一実施形態では、図２のＷｅｂサービスプラットフォーム２００が、読み取り要求及びトランザクション要求を受け取り、種々のデータベースインスタンスの種々のノードにそれらをルーティングしてもよい。なお、図１～５における上記の実施例では、データベース階層とストレージ階層とが分かれている場合について説明したが、他の例では、ストレージ階層は、データベースインスタンスから分かれていなくてもよい。さらに、他の実施例では、本システムはデータベースサービスでなくてもよく、その代わりに、記憶データの読み取り及び書き込みを実行可能な別のシステムであってもよい。さらになお、ライターである２つのノードは、両者の間でロックしていてもよい。ただし、同時処理を許可するために、リーダーであるノードとライターであるノードとは、両者の間でロックしていなくてもよい。

【００５２】

一実施形態では、様々なノードは、トランザクションの順序付けに用いられ得るクロックをノードごとに保持してもよい。これらのクロックは、ノード全体で同期されていてもよく、互いの範囲内で±δの精度を有し得る。これらのクロックの精度は、マルチノードシステムにおいてゼロではない場合がある。そのため、イベントがほぼ同時に発生する可能性があると、これらのイベントにおいて正確に順序付けを行うこと（因果関係）及び特定の分離レベルを維持することが難しくなる場合がある。例えば、読み取り一貫性の分離レベルには、以下の特性が含まれ得る。すなわち、ステートメントの開始時において、別のノードからコミットされたものは何でもその時点において見える時点がある、コミットされていない別のノードからの変更はその時点において見えない、及びコミットされていようとなかろうと、そのノード自体からの変更を見ることができる、という特性である。２つの時刻、すなわちＡとＢを所与として、次の３つのシナリオが発生し得る。Ａ＜Ｂ（Ｂの前にＡが起こる）、Ａ＞Ｂ（Ｂの後にＡが起こる）、及びＡ　Ｂ（ＡとＢがほぼ同時に起こり、ＡとＢが精度ウィンドウδ内にあるようになっている）。

【００５３】

いくつかの実施形態では、これらのクロックによって保持される時刻は、タイムスタンプ（例えば、２０１２年１５日、２０：００．３５　ＧＭＴ）であってもよく、他の実施形態では、これらの時刻は、ログシーケンス番号（ｌｏｇ　ｓｅｑｕｅｎｃｅ　ｎｕｍｂｅｒ：ＬＳＮ）などの、時刻を示す単調増加的な値であってもよい。この値は、システムのノードにわたる通信が発生すると増加し得る。ＬＳＮの実施例では、値が単調増加しているため、ＬＳＮ１００はＬＳＮ１０５よりも早い時点を示し得る。なお、ＬＳＮ時刻の間において各数値を用いる必要はない。従って、ＬＳＮ１００及び１０５は、一実施例で

は、最も密接して割り当てられた２つのＬＳＮであり得る。別の実施例では、ＬＳＮ１００～１０５のそれぞれを用いてもよい。

【００５４】

６２０に示すように、第１の時刻指示と第２の時刻指示とを、読み取りとトランザクションとにそれぞれ関連付けてもよい。例えば、いくつかの実施形態では、要求（例えば、トランザクション、読み取りなど）を受け取ると、その要求を受け取ったノードは、そのヘッドノードの個別のクロックに基づき、そのトランザクションに１つ以上の時刻を割り当ててもよい。例えば、ヘッドノードは、読み取り要求をＬＳＮ１００にて受け取ってもよく、一貫性の時点（スナップショット時刻）Ｔｓとして１００を割り当ててもよい。Ｔｓは、スナップショット時刻が作成された時を表し得る。別の実施例として、ヘッドノード（例えば、別のヘッドノード）は、テーブルを更新する要求をＬＳＮ１０１にて受け取ってもよい。上記ノードは、ＬＳＮ１０２にてテーブルを更新し、次いで、その更新をＬＳＮ１０３にてコミットしてもよい。このような実施例では、上記ノードは、トランザクションがコミットされた時を表すコミット時刻Ｔｃとして１０３を割り当ててもよい。なお、トランザクションに関連付けられた他の時刻を割り当ててもよい。例えば、Ｔｃに加えて、書き込みに関連付けられたＴｗが別の時刻にあってもよい。Ｔｗは、変更すべき最後のページがアンラッチ（例えば、ページの開放／アンロック）された直後の時刻を表し得るが、そのトランザクション要求を受け取ったヘッドノードによって割り当てられてもよい。別の実施例として、Ｔｓに加えて、読み取りに関連付けられたＴｒが別の時刻にあってもよい。Ｔｒは、最初のデータページが読み取りラッチされた直後の時刻を表し得るが、その読み取り要求を受け取ったヘッドノードによって割り当てられてもよい。

【００５５】

６３０に示すように、潜在的な読み取り異常が検出され得る。一実施形態では、読み取り要求（及びその読み取りの実行）を受け取ったヘッドノードは、ブロック６３０にて検出を実行し得る。いくつかの実施形態では、このような検出は、第１の時刻指示（例えば、Ｔｓ及び／またはＴｒ）が、第２の時刻指示（例えば、Ｔｃ及び／またはＴｗ）の閾値（例えば、精度ウィンドウδ）内にあるとの判定に基づき得る。各種の潜在的な読み取り異常が発生する可能性がある。例えば、ファジーリードは、更新及び（再）読み取りがほぼ同時に起こるときに発生する可能性があり、結果として同一レコードについて異なる値を読み取る可能性がある。δとして５を用いた表１の実施例では、読み取りの一貫性ポイントよりも前に起きたときにコミットを処理することにより、適切な値を読み取れるようにその読み取りが正しく動作する。表１の実施例を図７Ａに図示する。

【００５６】

【表１】

表１

| ノード１ | | ノード２ | |
|---|---|---|---|
| １０１ | トランザクション開始 | １００ | スナップショットの作成（$T_s$＝１００） |
| １０２ | 更新　Ｘ＝１０ | １０１ | |
| １０３ | コミット（$T_c$＝１０３） | １０２ | |
| １０４ | | １０３ | 読み取り　Ｘ（１０）［１００≒１０３］ |

【００５７】

ファジーリードを表２の実施例に示し、図７Ｂに図示する。これも同様に、δとして５を用いている。ファジーリードでは、リードトランザクションにより、ＬＳＮ１０１にて値（１）が読み取られ、次いで、同一レコードをＬＳＮ１０４にて再び読み取ると異なる値（１０）が見える。本明細書で説明されるように、開示された技術は、表２の潜在的なファジーリードを検出することができ、読み取りを調節することによってそのファジーリードが実際には発生しないようにすることができる。

【００５８】
【表２】

表２

| ノード１ | | ノード２ | |
|---|---|---|---|
| １０１ | トランザクション開始 | １００ | スナップショットの作成（$T_s$＝１００） |
| １０２ | | １０１ | 読み取り　Ｘ（１） |
| １０３ | 更新　Ｘ＝１０ | １０２ | |
| １０４ | コミット（$T_c$＝１０４） | １０３ | |
| １０５ | | １０４ | 読み取り　Ｘ（１０）［１００≒１０４］ |

【００５９】
なお、表２のファジーリードは、（例えば、（単数または複数の）クライアント上で）動作がシリアル化されている場合には発生しない。これは、更新を行ってから初めて行を読み取ることになるためである。
【００６０】
別の異常読み取りはリードスキューである。これは、（例えば、複数の異なるレコードの）一貫性のないデータを読み取った状況である。（図７Ｃに示した）表３の実施例について考察する。これも同様にδを５としている。表３及び図７Ｃの実施例では、データテーブルは、初期値をＸ＝１及びＹ＝２として、Ｘ＝２Ｙである不変関係を有する。図示の通り、この初期値１は、ＬＳＮ１０１にてＸについて読み取られるが、Ｙの更新値は、ＬＳＮ１０４にて２０と読み取られる。これは、Ｘ＝２Ｙと矛盾している。
【００６１】
【表３】

表３

| ノード１ | | ノード２ | |
|---|---|---|---|
| １０１ | トランザクション開始 | １００ | スナップショットの作成（$T_s$＝１００） |
| １０２ | 更新　Ｙ＝２０ | １０１ | 読み取り　Ｘ（１） |
| １０３ | 更新　Ｘ＝１０ | １０２ | |
| １０４ | コミット（$T_c$＝１０４） | １０３ | |
| １０５ | | １０４ | 読み取り　Ｙ（２０）［１００≒１０４］ |

【００６２】
いくつかの実施形態では、潜在的な読み取り異常は、読み取りの一貫性ポイントＴｓの精度ウィンドウ（δ）内にコミット時刻Ｔｃがある場合に検出され得る。潜在的な読み取り異常は、本明細書では、読み取り異常が発生し得る可能性が存在することを示すのに用いられるが、潜在的な読み取り異常は、読み取り異常が確実に発生することを必ずしも意味するものではないことに留意されたい。従って、読み取り異常の可能性が存在する場合、ブロック６４０にて以下で説明するように、システムは、それを検出し、潜在的な読み取り異常の回避を試みることができる。
【００６３】
上述したように、いくつかの実施形態では、Ｔｃ及びＴｓ以外の時刻も同様に、読み取り及び／または他のトランザクションに関連付けてもよい。例えば、Ｔｗを用いた実施形態では、Ｔｃ＞Ｔｓである場合、読み取りを行ってもトランザクションによってなされた変更が見えず、異常読み取りは発生しない。Ｔｃ＜Ｔｓである場合、読み取りを行うとトランザクションによってなされた変更が見える。Ｔｃ　Ｔｓかつ Ｔｗ＜Ｔｓである場合、トランザクションによってなされた変更が読み取りの前に行われていたため、潜在的な読

み取り異常は存在しない。トランザクションによってなされた変更は、読み取りを行うことによって見える。さもなければ、Ｔｃ　Ｔｓかつ Ｔｗ　Ｔｓの場合には、潜在的な読み取り異常が存在する。

【００６４】

いくつかの実施形態では、Ｔｒを用いて潜在的な読み取り異常を検出するようにしてもよい。このような実施形態では、Ｔｃ＞Ｔｓである場合、トランザクションによってなされた変更は読み取りを行っても見えず、異常読み取りは発生しない。Ｔｃ＜Ｔｓである場合、読み取りを行うとトランザクションによってなされた変更が見える。Ｔｃ　Ｔｓかつ Ｔｗ＜Ｔｒである場合、トランザクションによってなされた変更が読み取りの前に行われていたため、潜在的な読み取り異常は存在しない。さもなければ、Ｔｃ　Ｔｓかつ Ｔｗ　Ｔｒである場合には、潜在的な読み取り異常が存在し、ブロック６３０にて検出され得る。

【００６５】

なお、いくつかの実施形態では、リードスキュー及びファジーリードは、最初のページをラッチした後にのみ起こる場合がある。そのため、１つのレコードを検索するステートメントでは、こうした異常が発生しない。従って、いくつかの実施形態では、ブロック６３０の検出ロジックは、（同一レコードを複数回検索しようと複数の異なるレコードを検索しようと）複数のレコードを検索する場合にのみ実行されてもよい。

【００６６】

６４０に示すように、潜在的な読み取り異常を上記検出したことに応答して、その読み取り要求を受け取ったノードは、第１の時刻指示が第２の時刻指示よりも早い時点を示しているかどうかに関わらず、その読み取り要求によって指定された読み取りを、トランザクション要求によって指定されたトランザクションの後に実行させることができる。

【００６７】

いくつかの実施形態では、トランザクションの後に読み取りを実行させることは、その読み取りの再試行に関連付けられた再試行の時刻指示が第１の時刻指示よりも後の時刻を示すようにその読み取りを再試行する読み取り要求を受け取ったノードを含んでもよい。例えば、潜在的なファジーリードまたはリードスキューが検出された場合、読み取りステートメントは、Ｔｓを維持する一方でＴｒをリセットする（例えば、Ｔｒの時刻を進める）ことによって抽出され得る。再試行時にＴｓを維持することにより、フォワード進行を達成することができる。なぜなら、Ｔｒが進められるため、トランザクションと読み取りが、比較ロジックにとって既知の事例の１つ（例えば、潜在的な読み取り異常が発生しない状況）に結果として収まり得るためである。

【００６８】

なお、一実施形態では、再試行の場合、ブロック６３０の検出ロジックは、以前のＴｒを置き換えた再試行時刻（更新されたＴｒ）を用いて再度適用され得る。従って、その再試行が成功した場合、検出ロジックは、再試行された読み取りに対しては読み取り異常が発生しなかったと判定する。例えば、更新されたＴｒを用いることにより、異常読み取りが起こり得ない区分の１つ（例えば、Ｔｃ　Ｔｓかつ Ｔｗ＜Ｔｒ）の範囲内に再試行が収まっていると検出ロジックに判定させてもよく、異常読み取りを伴わずに再試行が行われる。一方、別の潜在的な異常読み取りが発生する場合があるため、別の再試行を行う際には、別の再試行の時刻を時間的にさらに進めるようにする。検出ロジックを適用し、潜在的な異常読み取りが存在すると判定し、そのステートメントを再試行することは、再試行が成功するまで何回でも発生する場合がある。あるいは、いくつかの実施形態では、ノードは、その読み取りが正確でない可能性があることを示すエラーメッセージを実際の読み取り値と共に返す前に、当該ステートメントをある最大回数（例えば、２回、３回、１０回など）再試行してもよい。

【００６９】

一実施形態では、この閾値を変更してもよい。例えば、システムにおいて既に発生したトランザクションに対する再試行の頻度に基づいて閾値を変更してもよい。ある実施例と

して、再試行が頻繁に発生している場合には、閾値を減少させて控えめの水準まで下げてもよい。同様に、閾値を増加させてもよい。

【００７０】

いくつかの実施形態では、ブロック６２０、６３０及び６４０は、同一レコードを複数回読み取ろうと複数の異なるレコードを読み取ろうと、複数のレコードの検索を含む読み取り要求に対して行われてもよい。従って、このような実施形態では、読み取り異常の有無の確認は、１つのレコードの検索に対してではなく、複数のレコードの検索に対して行われてもよい。そのため、第１の読み取りは制約のない読み取りとなり得るが、これは、１回の読み取りのみを実行中である場合、（上記のような読み取り異常と解釈するには、第１の読み取りと矛盾する第２の読み取りが必要となるため）リードスキューまたはファジーリードが起こる可能性がないためである。従って、いくつかの実施形態では、本システムは、互いの精度ウィンドウ（閾値）内にある、１つのレコードの検索を求める読み取り要求と、トランザクション要求とを処理する際に図６の方法を常に適用しなくてもよい。

【００７１】

いくつかの実施形態では、読み取り要求を受け取ったノードは、別のノードが書き込み要求を受け取って当該レコードを更新中であると認識する場合がある。例えば、本システムは、当該データへの変更を他のノードが認識することを保証する下位副構造を含んでもよい。例示的な下位副構造には、（例えば、各データページキャッシュ３３５または他のキャッシュに対するＷｅｂサービスプラットフォーム２００のレベルでの）キャッシュフュージョン、または共有ディスクが含まれる。一実施形態では、共通ストレージ（例えば、ストレージ階層）の上にコヒーレントキャッシュを置いてもよい。コヒーレントキャッシュを用いることにより、あるものをノードが書き込んだ場合、別のノードがそれを見ることが保証され得る。一実施形態では、様々な時刻指示がトランザクションテーブルに記憶されてもよい。例えば、Ｔｗは、コミットの時刻以降にトランザクションのコミット時刻と共に記憶されてもよく、各時刻の値を書き込み、トランザクション識別子を生成してもよい。トランザクション識別子は、所与のトランザクションがアクティブか、それともコミットされたかを示し得る。所与のレコードに対してトランザクションがアクティブである場合、そのレコードに対して読み取りを実行しているノードは、そのレコードに対して以前の値を生成し得る（例えば、ロールバック、ＵＮＤＯなど）。

【００７２】

本明細書で説明された方法は、様々な実施形態では、ハードウェアとソフトウェアの任意の組み合わせによって実装され得る。例えば、一実施形態では、本方法は、プログラム命令を実行する１つ以上のプロセッサを含むコンピュータシステム（例えば、図８と同様のコンピュータシステム）によって実行され得る。これらのプログラム命令は、上記プロセッサに接続されたコンピュータ可読記憶媒体に記憶されている。このプログラム命令は、本明細書で説明された機能（例えば、本明細書で説明されたサービス／システム及び／またはストレージサービス／ストレージシステムを実装する様々なサーバ及び他の構成要素の機能）を実装するように構成され得る。

【００７３】

開示されたトランザクションの順序付け技術を用いることにより、強力な、かつ理にかなった分離レベルを顧客に提供することができる。この技術により、システムのスケーラビリティを向上させることができる。なぜなら、スナップショット（読み取り）を迅速に作成することが可能であり、ネットワーク通信を必要としなくてもよいためである。さらに、トランザクションが重複しない作業負荷も適切にスケールアウトすることができる。シングルトン行の検索は迅速に行うことができる。というのも、それらの読み取りは、読み取り異常の検出ロジックを実行させずに制約なく行われ得るためである。（例えば、Ｔｃ及びＴｓに加えて、Ｔｒ及び／またはＴｗを計算に入れることによって）正確な精度ウィンドウを用いた実施形態では、ステートメントが再試行される可能性を減少させることができる。

【００７４】
図８は、種々の実施形態に係る、本明細書で説明されたシステムの少なくとも一部を実装するように構成されたコンピュータシステムを示すブロック図である。例えば、コンピュータシステム８００は、種々の実施形態では、（例えば、データベース階層もしくは同程度のシステムの）ノード、またはクライアントに代わってレコード及び関連メタデータを記憶する複数のストレージノードの１つを実装するように構成され得る。コンピュータシステム８００は、パーソナルコンピュータシステム、デスクトップコンピュータ、ラップトップコンピュータもしくはノートブックコンピュータ、メインフレームコンピュータシステム、ハンドヘルドコンピュータ、ワークステーション、ネットワークコンピュータ、コンシューマ機器、アプリケーションサーバ、ストレージ装置、電話、携帯電話、または一般的な任意の種類のコンピューティング装置を含む各種の装置のいずれでもあってよいが、これらに限定されることはない。
【００７５】
コンピュータシステム８００は、入出力（ｉｎｐｕｔ／ｏｕｔｐｕｔ：Ｉ／Ｏ）インターフェース８３０を介してシステムメモリ８２０に接続された１つ以上のプロセッサ８１０を含む（これらのプロセッサはいずれも複数のコアを含んでよく、それらのコアはシングルスレッド方式でもマルチスレッド方式でもよい）。コンピュータシステム８００は、Ｉ／Ｏインターフェース８３０に接続されたネットワークインターフェース８４０をさらに含む。様々な実施形態では、コンピュータシステム８００は、１つのプロセッサ８１０を含むユニプロセッサシステム、またはいくつかのプロセッサ８１０（例えば、２個、４個、８個もしくは他の適切な数）を含むマルチプロセッサシステムであってよい。プロセッサ８１０は、命令を実行可能な任意の適切なプロセッサであってよい。例えば、様々な実施形態では、プロセッサ８１０は、様々な命令セットアーキテクチャ（ｉｎｓｔｒｕｃｔｉｏｎ ｓｅｔ ａｒｃｈｉｔｅｃｔｕｒｅｓ：ＩＳＡ）のいずれかを実装する汎用プロセッサまたは組み込みプロセッサ（ｘ８６、ＰｏｗｅｒＰＣ、ＳＰＡＲＣもしくはＭＩＰＳのＩＳＡまたはその他の適切なＩＳＡなど）であってよい。マルチプロセッサシステムでは、プロセッサ８１０のそれぞれは、通常は同じＩＳＡを実装し得るが、必ずしもそのように実装されるとは限らない。コンピュータシステム８００は、１つ以上のネットワーク通信装置（例えば、ネットワークインターフェース８４０）も含み、他のシステム及び／または構成要素と通信ネットワーク（例えば、インターネット、ＬＡＮなど）を介して通信できるようになっている。例えば、システム８００上で実行されるクライアントアプリケーションは、ネットワークインターフェース８４０を用いて、本明細書で説明されたデータベースシステムの構成要素の１つ以上を実装する１つのサーバまたはサーバのクラスター上で実行されるサーバアプリケーションと通信してもよい。別の例では、コンピュータシステム８００上で実行されるサーバアプリケーションのインスタンスは、ネットワークインターフェース８４０を用いて、他のコンピュータシステム（例えば、コンピュータシステム８９０）に実装され得るサーバアプリケーション（または別のサーバアプリケーション）の他のインスタンスと通信してもよい。
【００７６】
例示した実施形態では、コンピュータシステム８００は、１つ以上の永続的記憶装置８６０及び／または１つ以上のＩ／Ｏ装置８８０も含む。様々な実施形態では、永続的記憶装置８６０は、ディスクドライブ、テープドライブ、ソリッドステートメモリ、他の大容量ストレージ装置、またはその他の永続的記憶装置に対応し得る。コンピュータシステム８００（または分散アプリケーションもしくはそのシステム上で動作するオペレーティングシステム）は、所望により、命令及び／またはデータを永続的記憶装置８６０に記憶してもよく、記憶された命令及び／またはデータを必要に応じて検索してもよい。例えば、ある実施形態では、コンピュータシステム８００は、ストレージシステムサーバノードをホスティングしてもよく、永続的記憶装置８６０は、そのサーバノードに付属するＳＳＤを含んでもよい。
【００７７】

コンピュータシステム８００は、１つ以上のシステムメモリ８２０を含んでおり、これらのメモリは、（単数または複数の）プロセッサ８１０によってアクセス可能な命令及びデータを記憶するように構成されている。様々な実施形態では、システムメモリ８２０は、任意の適切なメモリ技術（例えば、１つ以上のキャッシュ、ＳＲＡＭ（ｓｔａｔｉｃ ｒａｎｄｏｍ ａｃｃｅｓｓ ｍｅｍｏｒｙ）、ＤＲＡＭ、ＲＤＲＡＭ、ＥＤＯ ＲＡＭ、ＤＤＲ １０ ＲＡＭ、ＳＤＲＡＭ（ｓｙｎｃｈｒｏｎｏｕｓ ｄｙｎａｍｉｃ ＲＡＭ）、Ｒａｍｂｕｓ ＲＡＭ、ＥＥＰＲＯＭ、不揮発性／フラッシュ型メモリ、またはその他の種類のメモリ）を用いて実装され得る。システムメモリ８２０は、プログラム命令８２５を含み得る。これらの命令は、本明細書で説明された方法及び技術を実装するために、（単数または複数の）プロセッサ８１０によって実行可能である。様々な実施形態では、プログラム命令８２５は、プラットフォームのネイティブバイナリ、Ｊａｖａ（登録商標）バイトコードなどの任意のインタープリター型言語、もしくはＣ／Ｃ＋＋、Ｊａｖａ（登録商標）などのその他の言語、またはそれらの任意の組み合わせを用いてコード化され得る。例えば、プログラム命令８２５は、例示した実施形態では、データベース階層のデータベースエンジン・ヘッドノードの機能を実装するために実行可能なプログラム命令を含み、または、異なる実施形態では、データベース階層のクライアントに代わってデータベーステーブル及び関連メタデータを記憶する別個の分散データベース最適化ストレージシステムの複数のストレージノードのうちの１つの機能を実装するために実行可能なプログラム命令を含む。いくつかの実施形態では、プログラム命令８２５は、複数の独立したクライアント、サーバノード及び／または他の構成要素を実装してもよい。

【００７８】

いくつかの実施形態では、プログラム命令８２５は、オペレーティングシステム（図示せず）を実装するために実行可能な命令を含んでもよく、このオペレーティングシステムは、ＵＮＩＸ（登録商標）、ＬＩＮＵＸ（登録商標）、Ｓｏｌａｒｉｓ（登録商標）、ＭａｃＯＳ（登録商標）、Ｗｉｎｄｏｗｓ（登録商標）などの様々なオペレーティングシステムのいずれでもよい。プログラム命令８２５のいずれかまたは全ては、コンピュータプログラム製品、またはソフトウェアとして提供され得る。このコンピュータプログラム製品またはソフトウェアは、命令が記憶された非一時的コンピュータ可読記憶媒体を含んでもよく、この媒体を用いてコンピュータシステム（または他の電子装置）をプログラムすることにより、様々な実施形態に応じたプロセスを実行してもよい。非一時的コンピュータ可読記憶媒体は、マシン（例えば、コンピュータ）によって読み取り可能な形態（例えば、ソフトウェア、処理アプリケーション）で情報を記憶するための任意の機構を含んでもよい。一般的に言えば、非一時的コンピュータアクセス可能媒体は、コンピュータ可読記憶媒体またはメモリ媒体を含み得る。このような媒体としては、磁気媒体または光学媒体（例えば、Ｉ／Ｏインターフェース８３０を介してコンピュータシステム８００に接続されたディスクまたはＤＶＤ／ＣＤ－ＲＯＭ）などがある。非一時的コンピュータ可読記憶媒体は、ＲＡＭ（例えば、ＳＤＲＡＭ、ＤＤＲ ＳＤＲＡＭ、ＲＤＲＡＭ、ＳＲＡＭなど）、ＲＯＭなどの任意の揮発性または不揮発性媒体も含み得る。これらの媒体は、コンピュータシステム８００のいくつかの実施形態では、システムメモリ８２０または別の種類のメモリとして含まれ得る。他の実施形態では、プログラム命令は、ネットワークインターフェース８４０を介して実装され得るようなネットワーク及び／または無線リンクなどの通信媒体を経由して伝わる光学的、音響的または他の形態の伝搬信号（例えば、搬送波、赤外線信号、デジタル信号など）を用いて伝達され得る。

【００７９】

いくつかの実施形態では、システムメモリ８２０は、データ記憶部８４５を含んでもよく、この記憶部は、本明細書で説明されたように構成され得る。例えば、本明細書で説明されたデータベース階層の機能を実行する際に用いられるトランザクションログ、ＵＮＤＯログ、キャッシュ済みのページデータまたは他の情報などの、データベース階層によって（例えば、データベースエンジン・ヘッドノード上に）記憶されているものとして本明細書で説明された情報は、別々の時刻にて、及び様々な実施形態において、１つ以上のノ

ード上のデータ記憶部８４５もしくはシステムメモリ８２０の別の部分、永続的記憶装置８６０、及び／または１つ以上のリモートストレージ装置８７０に記憶され得る。同様に、データベース階層によって記憶されているものとして本明細書で説明された情報（例えば、本明細書で説明された分散ストレージシステムの機能を実行する際に用いられるＲＥＤＯログレコード、併合されたデータページ、及び／または他の情報など）は、別々の時刻にて、及び様々な実施形態において、１つ以上のノード上のデータ記憶部８４５もしくはシステムメモリ８２０の別の部分、永続的記憶装置８６０、及び／または１つ以上のリモートストレージ装置８７０に記憶され得る。概して、システムメモリ８２０（例えば、システムメモリ８２０内のデータ記憶部８４５）、永続的記憶装置８６０及び／またはリモートストレージ８７０は、データブロック、データブロックのレプリカ、データブロック及び／もしくはそれらの状態に関連付けられたメタデータ、データベース構成情報、並びに／または本明細書で説明された方法及び技術を実装するのに使用可能なその他の情報を記憶し得る。

【００８０】

一実施形態では、Ｉ／Ｏインターフェース８３０は、ネットワークインターフェース８４０または他の周辺インターフェースを通るものを含む、プロセッサ８１０、システムメモリ８２０及びシステム内の任意の周辺装置の間のＩ／Ｏトラフィックを調整するように構成され得る。いくつかの実施形態では、Ｉ／Ｏインターフェース８３０は、任意の必要なプロトコル変換、タイミング変換または他のデータ変換を実行することにより、ある構成要素（例えば、システムメモリ８２０）からのデータ信号を別の構成要素（例えば、プロセッサ８１０）での使用に適した形式に変換し得る。いくつかの実施形態では、Ｉ／Ｏインターフェース８３０は、例えば、ＰＣＩ（Ｐｅｒｉｐｈｅｒａｌ Ｃｏｍｐｏｎｅｎｔ Ｉｎｔｅｒｃｏｎｎｅｃｔ）バス規格またはＵＳＢ（Ｕｎｉｖｅｒｓａｌ Ｓｅｒｉａｌ Ｂｕｓ）規格を変型したものなどの様々な種類の周辺バスを介して接続された装置にも対応し得る。いくつかの実施形態では、Ｉ／Ｏインターフェース８３０の機能は、例えば、ノースブリッジ及びサウスブリッジなどの、２つ以上の別々の構成要素に分割され得る。また、いくつかの実施形態では、システムメモリ８２０とのインターフェースなどの、Ｉ／Ｏインターフェース８３０の機能の一部または全てをプロセッサ８１０に直接組み込んでもよい。

【００８１】

ネットワークインターフェース８４０は、例えば、コンピュータシステム８００と、他のコンピュータシステム８９０（このシステムは、本明細書で説明されたデータベースシステムの１つ以上のストレージシステムサーバノード、データベースエンジン・ヘッドノード及び／またはクライアントを実装し得る）などの、ネットワークに接続された他の装置との間でデータを交換できるように構成され得る。加えて、ネットワークインターフェース８４０は、コンピュータシステム８００と、各種Ｉ／Ｏ装置８５０及び／またはリモートストレージ８７０との間の通信を行うことができるように構成され得る。入出力装置８５０は、いくつかの実施形態では、１つ以上の表示端末、キーボード、キーパッド、タッチパッド、スキャニング装置、音声認識装置もしくは光学的認識装置、または１つ以上のコンピュータシステム８００によるデータの入力もしくは検索に適したその他の装置を含んでもよい。複数の入出力装置８５０は、コンピュータシステム８００内に存在してもよく、またはコンピュータシステム８００を含む分散システムの様々なノードに分散されてもよい。いくつかの実施形態では、同様の入出力装置は、コンピュータシステム８００から分かれていてもよく、コンピュータシステム８００を含む分散システムの１つ以上のノードと、ネットワークインターフェース８４０を介するような有線または無線接続を通じて相互作用してもよい。ネットワークインターフェース８４０は、通常１つ以上の無線ネットワーク接続プロトコル（例えば、Ｗｉ－Ｆｉ／ＩＥＥＥ８０２．１１、または別の無線ネットワーク接続規格）に対応し得る。しかしながら、様々な実施形態では、ネットワークインターフェース８４０は、例えば、他の種類のイーサネット（登録商標）ネットワークなどの、任意の適切な有線または無線の一般的なデータネットワークを経由した通

信に対応し得る。さらに、ネットワークインターフェース８４０は、アナログ音声ネットワークもしくはデジタルファイバー通信ネットワークなどの遠距離通信ネットワーク／電話網、ファイバーチャネルＳＡＮなどのストレージエリアネットワーク、またはその他の適切な種類のネットワーク及び／またはプロトコルを経由した通信に対応し得る。様々な実施形態では、コンピュータシステム８００は、図８に示した構成要素よりも多い構成要素、これよりも少ない構成要素、またはこれとは異なる構成要素（例えば、ディスプレイ、ビデオカード、オーディオカード、周辺装置、ＡＴＭインターフェース、イーサネットインターフェース、フレームリレーインターフェースなどの他のネットワークインターフェース）を含んでもよい。

【００８２】

なお、本明細書で説明された分散システムの任意の実施形態、またはそれらの任意の構成要素は、１つ以上のＷｅｂサービスとして実装され得る。例えば、データベースシステムのデータベース階層内のデータベースエンジン・ヘッドノードは、本明細書で説明された分散ストレージシステムを利用するデータベースサービス及び／または他の種類のデータストレージサービスを、Ｗｅｂサービスとしてクライアントに提供し得る。いくつかの実施形態では、Ｗｅｂサービスは、ネットワークを介して相互運用可能なマシン対マシン相互作用に対応するように設計されたソフトウェア及び／またはハードウェアシステムによって実装され得る。Ｗｅｂサービスは、Ｗｅｂサービス記述言語（Ｗｅｂ　Ｓｅｒｖｉｃｅｓ　Ｄｅｓｃｒｉｐｔｉｏｎ　Ｌａｎｇｕａｇｅ：ＷＳＤＬ）などの、マシン処理可能な書式を用いて記述されたインターフェースを備えてもよい。他のシステムは、Ｗｅｂサービスのインターフェースを記述することによって定められる方法でＷｅｂサービスと相互作用してもよい。例えば、Ｗｅｂサービスは、他のシステムが呼び出すことができる様々な動作を規定してもよく、様々な動作を要求する際に他のシステムが適合していると見込むことができる特定のアプリケーションプログラミングインターフェース（ＡＰＩ）を規定してもよい。

【００８３】

上記は、前述の条項に照らしてより理解され得る。

条項１

システムであって、

それぞれが少なくとも１つのプロセッサ、及びメモリを含む複数のコンピューティングノードであって、データベースサービスを集合的に実施するように構成されている複数のコンピューティングノード、を含み、

前記複数のコンピューティングノードの第１のノードは、

データベーステーブル内の特定のデータページの特定のデータレコードを対象とする書き込み要求であって、前記特定のデータレコードに対して行うべき変更を指定する書き込み要求を受け取り、

前記変更をコミットする時刻を示すコミット時刻を前記書き込み要求に割り当てるように構成されており、

前記複数のコンピューティングノードの第２のノードは、

前記データベーステーブル内の前記特定のデータページを読み取る読み取り要求を受け取り、

読み取りの一貫性ポイントを示す読み取り一貫性ポイント時刻を前記読み取り要求に割り当て、

前記読み取り一貫性ポイント時刻が前記コミット時刻の精度ウィンドウ内にあると判定し、

前記読み取り一貫性ポイント時刻が前記コミット時刻よりも早い時点を示しているかどうかに関わらず、前記書き込み要求によって指定された前記変更がコミットされた後に前記読み取り要求の実行を行わせるように構成されている、システム。

条項２

前記複数のコンピューティングノードの前記第２のノードは、

前記書き込み要求に関連付けられた書き込みアンラッチ時刻が、前記読み取り要求に関連付けられた読み取りラッチ時刻の精度ウィンドウ内にあると判定するようにさらに構成されており、
前記書き込み要求によって指定された前記変更がコミットされた後に前記読み取り要求の実行を前記行わせることは、
前記読み取りラッチ時刻を後の時点に移動すること、及び
前記読み取り一貫性の時点以降に前記読み取り要求を再試行すること、を含む条項１に記載のシステム。
条項３
前記書き込み要求によって指定された前記変更がコミットされた後に前記読み取り要求の実行を前記行わせることは、前記移動された読み取りラッチ時刻が前記書き込みアンラッチ時刻の精度ウィンドウ内にはないと判定すること、をさらに含む、条項２に記載のシステム。
条項４
前記第１のノード及び前記第２のノードのそれぞれは、互いの前記精度ウィンドウ内に維持されている個別のクロックを保持するように構成されており、前記コミット時刻が前記第１のノードによって決定され、前記読み取り一貫性の時点が前記第２のノードによって決定される、条項１に記載のシステム。
条項５
方法であって、
複数のコンピューティングノードによって、
１つ以上のクライアントから、記憶されたレコードの読み取りを実行する読み取り要求、及び前記レコードの更新を実行する更新要求を受け取ること、
第１の時刻指示と第２の時刻指示とを、前記読み取りと前記更新とにそれぞれ関連付けること、並びに
前記第１の時刻指示が前記第２の時刻指示の閾値内にあると判定したことに少なくとも部分的に基づき、潜在的な読み取り異常を検出すること、
を実行すること、を含む方法。
条項６
前記潜在的な読み取り異常を前記検出したことに応答して、前記第１の時刻指示が前記第２の時刻指示よりも早い時点を示しているかどうかに関わらず、前記更新要求によって指定された前記更新の後に前記読み取り要求によって指定された前記読み取りを実行させること、をさらに含む条項５に記載の方法。
条項７
前記読み取りの再試行に関連付けられた再試行の時刻指示が前記第１の時刻指示よりも後の時刻を示すように、前記第１の時刻指示以降に前記読み取りを再試行すること、及び
前記再試行された読み取りに対して読み取り異常が発生していないと判定すること、をさらに含む条項５に記載の方法。
条項８
前記第２の時刻指示は前記更新をコミットする時刻を示す、条項５に記載の方法。
条項９
前記潜在的な読み取り異常を前記検出することは、第３の時刻指示にさらに基づいており、前記第３の時刻指示も前記更新要求に関連付けられている、条項５に記載の方法。
条項１０
前記潜在的な読み取り異常を前記検出することは、前記第３の時刻指示が第４の時刻指示の閾値内にあると判定したことにさらに基づいており、前記第４の時刻指示も前記読み取り要求に関連付けられている、条項９に記載の方法。
条項１１
前記読み取り要求は前記複数のノードの第１のノードによって受け取られ、前記更新要求は前記複数のノードの別の第２のノードによって受け取られ、前記第１のノード及び前

記第２のノードは個別のクロックをそれぞれ保持しており、前記読み取りに関連付けられた前記第１の時刻指示は前記第１のノードの前記個別のクロックによって決定され、前記更新に関連付けられた前記第２の時刻指示は前記第２のノードの前記個別のクロックによって決定される、条項５に記載の方法。

条項１２

前記潜在的な読み取り異常は潜在的なファジーリードである、条項５に記載の方法。

条項１３

前記潜在的な読み取り異常は潜在的なリードスキューである、条項５に記載の方法。

条項１４

再試行の頻度に少なくとも部分的に基づいて前記閾値を変更すること、をさらに含む条項５に記載の方法。

条項１５

前記検出することは、前記レコードの前記読み取り、及び別のレコードの読み取りまたは前記レコードの第２の読み取りのいずれかを含む読み取り要求に対して実行される、条項５に記載の方法。

条項１６

データベースサービスのデータベースノードを実装するためにコンピュータで実行可能なプログラム命令を記憶する非一時的コンピュータ可読記憶媒体であって、前記データベースノードは、

前記データベースサービスによって記憶されたレコードの読み取りを指定する読み取り要求によって指定された前記読み取りに第１の時刻指示を関連付け、

前記第１の時刻指示が、別のデータベースノードによって受け取られたトランザクション要求であって、前記レコードを変更するトランザクションを指定するトランザクション要求によって指定された前記トランザクションに関連付けられた第２の時刻指示の閾値内にあると判定し、

前記第１の時刻指示が前記第２の時刻指示よりも早い時点を示しているかどうかに関わらず、前記トランザクション要求によって指定された前記変更がコミットされた後に前記読み取り要求の実行を行わせるように構成されている、非一時的コンピュータ可読記憶媒体。

条項１７

前記データベースノードは、

第３の時刻指示を前記読み取りに関連付け、

前記第３の時刻指示が、前記トランザクションに関連付けられた第４の時刻指示の閾値内にあると判定するようにさらに構成されており、

前記トランザクション要求によって指定された前記変更がコミットされた後に前記読み取り要求の実行を前記行わせることは、

前記第３の時刻指示を後の時点に置き換えること、及び

前記第１の時刻指示以降に前記読み取り要求を再試行すること、を含む条項１６に記載の非一時的コンピュータ可読記憶媒体。

条項１８

前記データベースノードは、前記後の時点が、前記第４の時刻指示の閾値内にはないと判定するようにさらに構成されている、条項１７に記載の非一時的コンピュータ可読記憶媒体。

条項１９

前記第１の時刻指示及び前記第２の時刻指示は、単調増加する時刻指標である、条項１６に記載の非一時的コンピュータ可読記憶媒体。

条項２０

前記読み取り要求は、前記レコードを含む複数のレコードを検索する要求である、条項１６に記載の非一時的コンピュータ可読記憶媒体。

【００８４】

様々な実施形態では、Ｗｅｂサービスは、Ｗｅｂサービス要求に関連付けられたパラメータ及び／またはデータを含むメッセージの利用を通じて要求または起動され得る。こうしたメッセージは、ＸＭＬ（Ｅｘｔｅｎｓｉｂｌｅ Ｍａｒｋｕｐ Ｌａｎｇｕａｇｅ）などの特定のマークアップ言語に従ってフォーマットされてもよく、かつ／またはＳＯＡＰ（Ｓｉｍｐｌｅ Ｏｂｊｅｃｔ Ａｃｃｅｓｓ Ｐｒｏｔｏｃｏｌ）などのプロトコルを用いてカプセル化されてもよい。Ｗｅｂサービス要求を実行するために、Ｗｅｂサービスクライアントは、その要求を含むメッセージを集めてもよく、そのメッセージを、ＨＴＴＰ（ＨｙｐｅｒＴｅｘｔ Ｔｒａｎｓｆｅｒ Ｐｒｏｔｏｃｏｌ）などの、インターネットを利用したアプリケーション層の転送プロトコルを用いて、Ｗｅｂサービスに対応するアドレス可能エンドポイント（例えば、ＵＲＬ（Ｕｎｉｆｏｒｍ Ｒｅｓｏｕｒｃｅ Ｌｏｃａｔｏｒ））に伝達してもよい。

【００８５】

いくつかの実施形態では、Ｗｅｂサービスは、メッセージを利用した技術ではなく、「ＲＥＳＴｆｕｌ（Ｒｅｐｒｅｓｅｎｔａｔｉｏｎａｌ Ｓｔａｔｅ Ｔｒａｎｓｆｅｒ）」技術を用いて実装されてもよい。例えば、ＲＥＳＴｆｕｌ技術に従って実装されたＷｅｂサービスは、ＳＯＡＰメッセージ内にカプセル化されたものではなく、ＰＵＴ、ＧＥＴまたはＤＥＬＥＴＥなどのＨＴＴＰメソッド内に含まれるパラメータを通じて起動され得る。

【００８６】

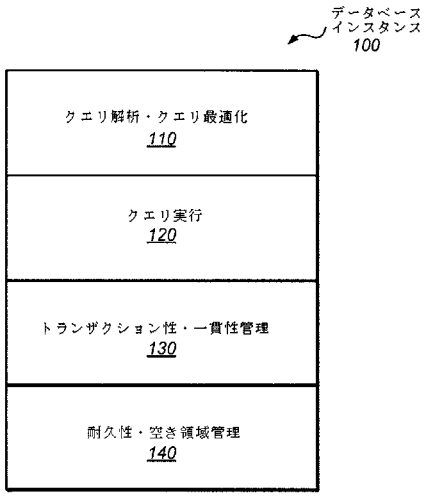
各図面に図示され、本明細書で説明された様々な方法は、方法の例示的な実施形態を表す。これらの方法は、ソフトウェア、ハードウェアまたはそれらの組み合わせを用いて手作業で実施されてもよい。あらゆる方法の順序は変更可能であり、様々な要素の追加、並べ替え、組み合わせ、省略、修正などを行ってもよい。

【００８７】

上記のような実施形態についてかなり詳細に説明してきたが、上記の開示を一旦完全に理解すれば、当業者にとって明らかとなるような多くの変形及び修正をなすことが可能である。以下の特許請求の範囲は、このような変形及び修正の全て、並びに従って、限定的な意味ではなく例示とみなされるべき上述の説明を包含するように解釈されることが意図される。

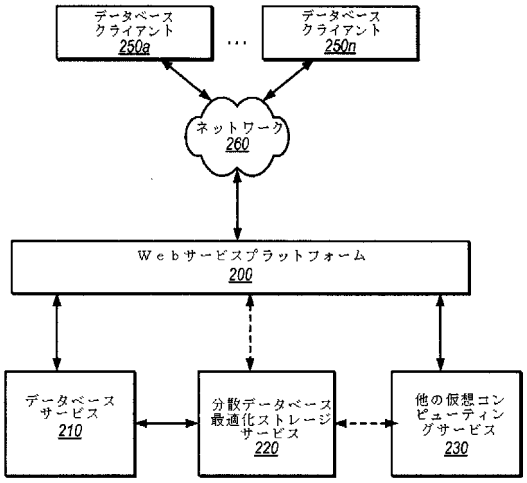
【図１】

図１



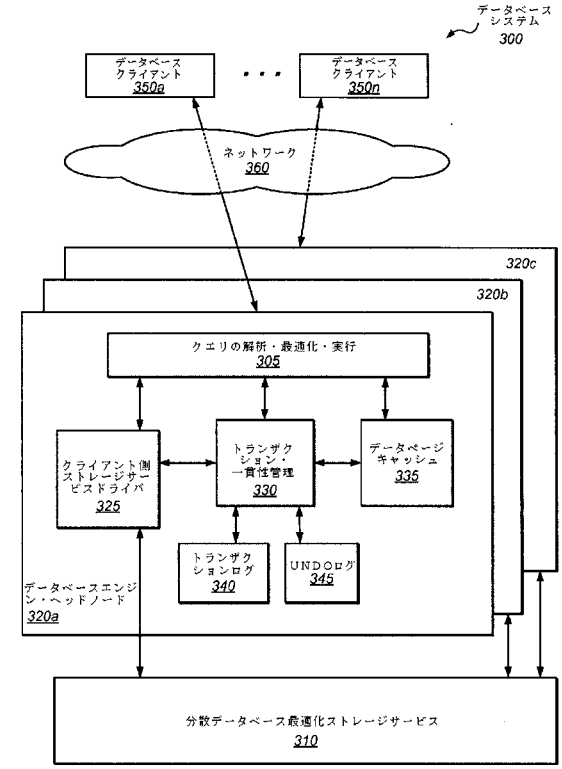

【図２】

図２



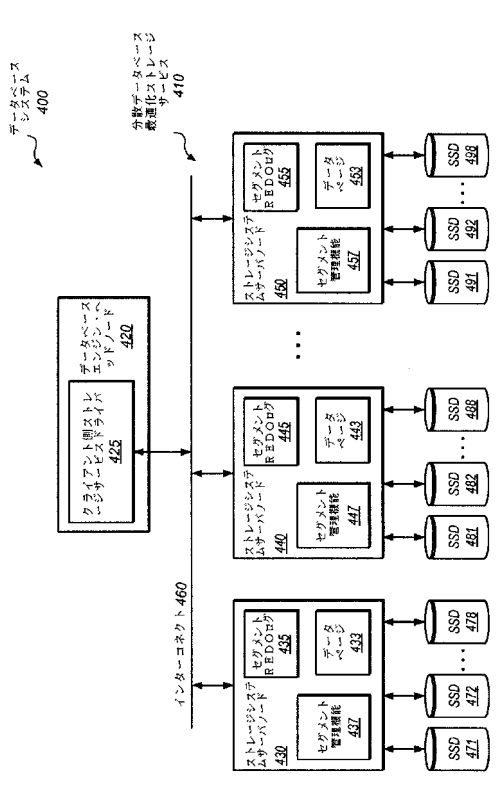

【図３】

図３



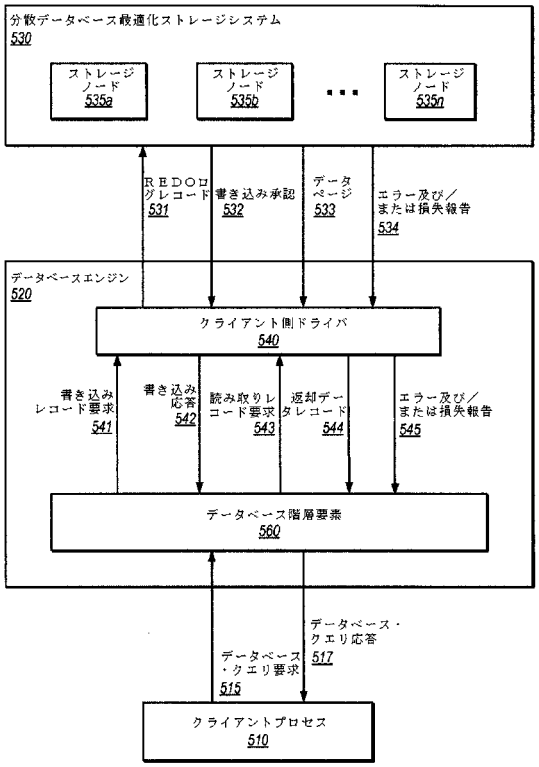

【図４】

図４



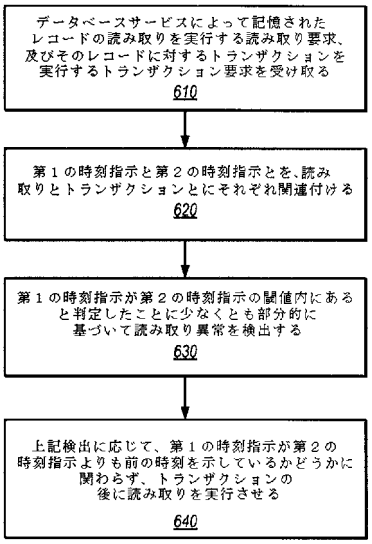
【図５】

図５




【図６】

図６




【図７】

図７Ａ

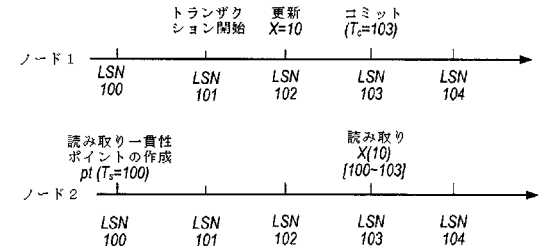



図７Ｂ

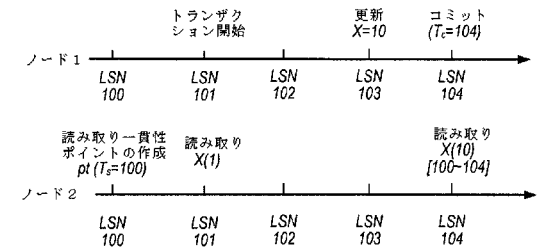
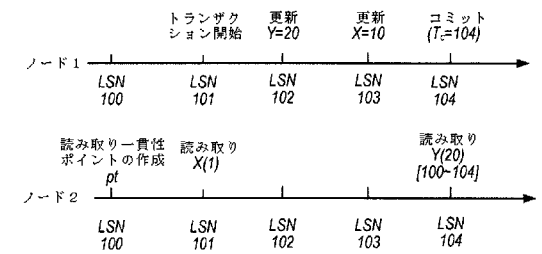
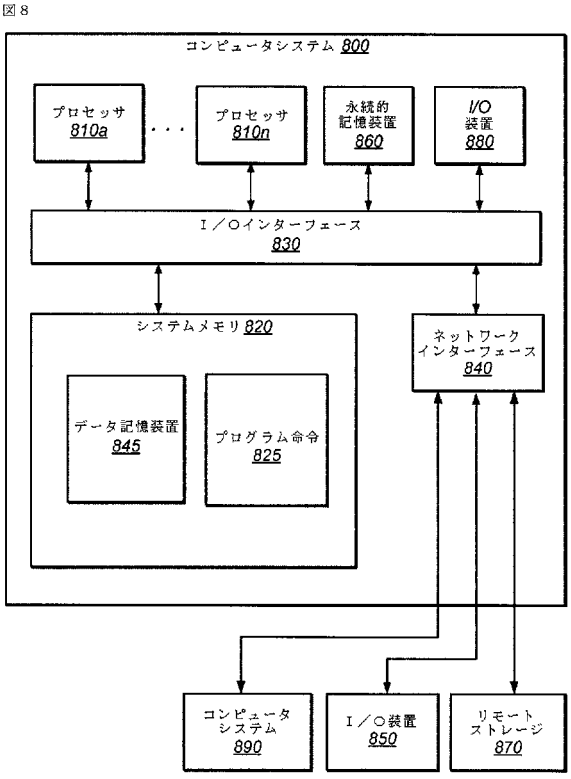



図７Ｃ




【図８】

図８



【国際調査報告】

## INTERNATIONAL SEARCH REPORT

International application No.
PCT/US 14/37901

A. CLASSIFICATION OF SUBJECT MATTER
IPC(8) - G06F 17/00 (2014.01)
CPC - G06F 2201/84
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)
CPC: G06F 2201/84; IPC(8): G06F 17/00 (2014.01)

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
CPC: Y10S 707/99954, G06F 17/30306, Y10S 707/99942, G06F 11/1464, G06F 11/1469, G06F 11/1435, G06F 2201/84, G06F 17/30067, G06F 11/1451; IPC(8): G06F 17/00 (2014.01) (keyword limited)

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
PatBase, Google Patents, IEEE; Search Terms: transaction ordering, database, nodes, record read, commit, consistency point, write read request, latch time, accuracy window, potential read anomaly, deadlock detection, latch, unlatch

C. DOCUMENTS CONSIDERED TO BE RELEVANT

| Category* | Citation of document, with indication, where appropriate, of the relevant passages | Relevant to claim No. |
|---|---|---|
| X | US 5,452,445 A (Hallmark et al.) 19 September 1995 (19.09.1995), entire document especially abstract, col. 8, ln 24-38, col. 9, ln 15 - col. 10, ln 17, Figure 2, steps 201 - 204, Figure 3, step 301 - 304, Figure 4, steps 401 - 407 | 1 - 15 |
| A | US 2007/0130238 A1 (Harris et al.) 07 June 2007 (07.06.2007), entire document | 1 - 15 |
| A | US 2006/0112222 A1 (Barrall) 25 May 2006 (25.05.2006), entire document | 1 - 15 |
| A | US 7,707,219 B1 (Bruso et al.) 27 April 2010 (27.04.2010), entire document | 1 - 15 |
| A | US 2005/0066095 A1 (Mullick et al.) 24 March 2005 (24.03.2005), entire document | 1 - 15 |

☐ Further documents are listed in the continuation of Box C. ☐

* 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 application or patent 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
"&" document member of the same patent family

| Date of the actual completion of the international search | Date of mailing of the international search report |
|---|---|
| 28 August 2014 (28.08.2014) | 03 OCT 2014 |
| Name and mailing address of the ISA/US<br>Mail Stop PCT, Attn: ISA/US, Commissioner for Patents<br>P.O. Box 1450, Alexandria, Virginia 22313-1450<br>Facsimile No. 571-273-3201 | Authorized officer:<br>Lee W. Young<br>PCT Helpdesk: 571-272-4300<br>PCT OSP: 571-272-7774 |

フロントページの続き

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

(72)発明者 マダヴァラプ，プラディープ・ジュニャーナ
アメリカ合衆国・９８１０９－５２１０・ワシントン州・シアトル・テリー アヴェニュ ノース・４１０
(72)発明者 ニューコム，クリストファー・リチャード
アメリカ合衆国・９８１０９－５２１０・ワシントン州・シアトル・テリー アヴェニュ ノース・４１０
(72)発明者 グプタ，アヌラグ・ウィンドラス
アメリカ合衆国・９８１０９－５２１０・ワシントン州・シアトル・テリー アヴェニュ ノース・４１０