(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第6522812号
(P6522812)
(45)発行日　令和1年5月29日(2019.5.29)　(24)登録日　令和1年5月10日(2019.5.10)

(51)Int.Cl.　F I
G06F 12/00 (2006.01)　G06F 12/00 531R
G06F 12/00 545A
G06F 12/00 531J

請求項の数 15　外国語出願　(全 51 頁)

(21)出願番号　特願2018-1662(P2018-1662)
(22)出願日　平成30年1月10日(2018.1.10)
(62)分割の表示　特願2016-501614(P2016-501614)の分割
原出願日　平成26年3月12日(2014.3.12)
(65)公開番号　特開2018-77895(P2018-77895A)
(43)公開日　平成30年5月17日(2018.5.17)
審査請求日　平成30年1月10日(2018.1.10)
(31)優先権主張番号　61/799,609
(32)優先日　平成25年3月15日(2013.3.15)
(33)優先権主張国　米国(US)
(31)優先権主張番号　14/201,505
(32)優先日　平成26年3月7日(2014.3.7)
(33)優先権主張国　米国(US)

(73)特許権者　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・９８１０８－１２２６・ワシントン州・シアトル・ピイオーボックス・８１２２６
(74)代理人　100098394
弁理士　山川　茂樹
(74)代理人　100064621
弁理士　山川　政樹
(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

最終頁に続く

(54)【発明の名称】分散型データベースシステム用高速クラッシュ回復

(57)【特許請求の範囲】
【請求項１】
分散型ストレージシステムを実装する複数のストレージノードであって、前記分散型ストレージシステムは、データベースのためにログ構造化データストレージを実装するように構成され、前記複数のストレージノードは、ネットワークを通じてデータベースシステムに接続し、前記ログ構造化データストレージの複数のリドゥログレコードが、前記複数のストレージノードにおいて、前記ネットワークを通じて前記データベースシステムから既に受信されており、前記複数のリドゥログレコードのそれぞれは、前記複数のストレージノードの中で前記データベースのために記憶されたデータに対する変更を記述する、複数のストレージノードと、
前記データベースシステムを実装するデータベースヘッドノードであって、前記データベースヘッドノードは、
前記ネットワークを通じて、前記複数のストレージノードとの接続を確立することと、
前記リドゥログレコードをリプレイすることなく、一つ以上のアクセス要求のための前記データベースへのアクセスを提供することと、
を含む、データベースヘッドノード故障からの回復を実行する、データベースヘッドノードと、
を備えるシステム。
【請求項２】

前記データベースヘッドノードはさらに、
前記データベースに対するアクセス要求を受信するように、
前記受信したアクセス要求に基づいて、前記ストレージノードに記憶されたデータページの現在の状態に対する要求を、前記ネットワークを通じて、前記複数のストレージノードのうちの一つに送信するように、及び
前記ストレージノードにおいて、その現在の状態で前記データページを生成するために、前記複数のリドゥログレコードのうちの一つ以上が、前記データページの既に保存された状態に適用された前記要求されたデータページを、その現在の状態で前記ストレージノードから受信するように構成される、請求項１に記載のシステム。
【請求項３】
前記複数のストレージノードに送信された前記複数のリドゥログレコードのうちの少なくともいくつかはシステムトランザクションを備え、前記複数のストレージノードのうちの一つのストレージノードは、
前記システムトランザクションが不完全であると決定するように、及び
前記少なくともいくつかのリドゥログレコードによって変更されたデータページの現在の状態を生成するときに適用されないとして前記複数のリドゥログレコードのうちの前記少なくともいくつかを識別するように、
構成される、請求項１に記載のシステム。
【請求項４】
前記複数のストレージノードに送信された前記複数のリドゥログレコードのうちの少なくともいくつかはシステムトランザクションを備え、前記データベースヘッドノードはさらに、
前記システムトランザクションが不完全であると決定するように、
前記少なくともいくつかのリドゥログレコードによって変更されたデータページの現在の状態を生成するときに適用されないとして前記複数のリドゥログレコードのうちの少なくともいくつかを識別するように、及び
前記複数のストレージノードのうちの一つ以上に、適用されない、前記複数のリドゥログレコードのうちの前記識別された少なくともいくつかを示す通知を送信するように、
構成される、請求項１に記載のシステム。
【請求項５】
データベースヘッドノードを実装する一つ以上のコンピューティング装置によって、
データベースヘッドノード故障からの回復時に、
データベースのためのデータを記憶する分散型ストレージシステムを実装する複数のストレージノードの一つ以上のストレージノードとの接続を確立することであって、前記複数のストレージノードは、前記データベースヘッドノードを実装する前記一つ以上のコンピューティング装置以外の複数のコンピューティング装置に実装され、前記分散型ストレージシステムは、前記データベースのためにログ構造化データストレージを実装するように構成され、前記ログ構造化データストレージの複数のリドゥログレコードが、前記複数のストレージノードにおいて既に受信されており、前記複数のリドゥログレコードのそれぞれは、前記データベースのために記憶されたデータに対する変更を、それが受信された前記それぞれのストレージノードで記述する、接続を確立することと、
前記リドゥログレコードのリプレイなしで、一つ以上のアクセス要求のための前記データベースへのアクセスを提供することと、
を実行することを含む方法。
【請求項６】
前記データベースに対するアクセス要求を受信することと、
前記アクセス要求を受信することに応えて、前記一つ以上のストレージノードから、前記データベースのための前記データの部分を記憶する一つ以上のデータページの現在の状態を要求することと、
前記アクセス要求にサービスを提供するために、前記データページのための前記データ

の部分を記憶する前記一つ以上のストレージノードから前記一つ以上のデータページの現在の状態を受信することと、
をさらに含む、請求項５に記載の方法。
【請求項７】
受信された前記データベースのための前記データの部分を記憶する前記一つ以上のデータページのうちの少なくとも一つの前記現在の状態は、前記一つ以上のストレージノードのうちの一つが、前記少なくとも一つのデータページの既に記憶されたバージョンまで前記複数のリドゥログレコードのうちの一つ以上をリプレイすることによって生成される、請求項６に記載の方法。
【請求項８】
受信された前記データベースのための前記データの部分を記憶する前記一つ以上のデータページのうちの前記少なくとも一つからの異なるデータページの前記現在の状態は、前記データページの既に記憶されたバージョンまで前記複数のリドゥログレコードのうちの一つ以上をリプレイすることなく、前記一つ以上のストレージノードのうちの一つによって送信される、請求項７に記載の方法。
【請求項９】
前記データベースヘッドノードは、前記複数のストレージノードにリドゥログレコードとして送信された変更をアンドゥするために複数のアンドゥログレコードを維持し、前記方法は、
前記一つ以上のストレージノードから受信した前記一つ以上のデータページのうちの一つが、不完全なユーザトランザクションによって影響を及ぼされると決定することであって、前記ユーザトランザクションは、前記一つのデータページを含む前記一つ以上のストレージノードに記憶された前記データに対して変更を指示することと、
前記ユーザトランザクションによって前記データページに対して指示された変更をアンドゥするために、前記データページに一つ以上のアンドゥログレコードを適用することと、
をさらに含む、請求項６に記載の方法。
【請求項１０】
前記データベースヘッドノードは、前記ユーザトランザクションを含む複数の不完全なユーザトランザクションを示すトランザクションテーブルを維持し、前記方法は、
前記トランザクションテーブルに少なくとも部分的に基づいて、前記複数の不完全なユーザトランザクションのうちの少なくとも一つによって影響を及ぼされた一つ以上の追加のデータページを決定することと、
前記一つ以上のストレージノードから、一つ以上の追加のデータページの現在の状態を要求することと、
前記一つ以上の追加のデータページを受信することに応えて、追加の一つ以上のアンドゥログレコードを前記一つ以上の追加のデータページに適用して、前記少なくとも一つの不完全なユーザトランザクションによって前記一つ以上の追加のデータページに対して指示された変更をアンドゥすることと、
をさらに含む、請求項９に記載の方法。
【請求項１１】
前記一つ以上の追加のデータページを決定することと、前記一つ以上の追加のデータページを要求することと、及び前記一つ以上の追加のデータページに前記追加の一つ以上のアンドゥログレコードを適用することは、前記データベースヘッドノードでバックグラウンドプロセスの一部として実行され、また、前記アクセス要求を受信することと、前記一つ以上のデータページの前記現在の状態を要求することと、及び前記一つ以上のデータページの現在の状態を受信することは、フォアグラウンドプロセスの一部として実行される、請求項１０に記載の方法。
【請求項１２】
前記データベースヘッドノード故障からの回復時に、

アクセスのために前記データベースを利用可能にする前に、前記データベースのための前記複数のストレージノードに記憶された前記データの、既に記録されたスナップショットに対応する状態への復元に対する要求を前記複数のストレージノードに送信することであって、前記復元が、前記複数のリドゥログのうちの一つ以上を、前記データの以前のバージョンに適用することを含むことと、
をさらに含む、請求項５に記載の方法。
【請求項１３】
一つ以上のコンピューティング装置による実行時に、
データベースヘッドノード故障からの回復時に、
データベースのためのデータを記憶する分散型ストレージシステムを実装する複数のストレージノードの一つ以上のストレージノードとの接続を確立することであって、前記複数のストレージノードは、前記データベースヘッドノードを実装する前記一つ以上のコンピューティング装置以外の複数のコンピューティング装置に実装され、前記分散型ストレージシステムは、前記データベースのためにログ構造化データストレージを実装するように構成され、前記ログ構造化データストレージの複数のリドゥログレコードが、前記複数のストレージノードにおいて既に受信されており、前記複数のリドゥログレコードのそれぞれは、前記データベースのために記憶されたデータに対する変更を、それが受信された前記それぞれのストレージノードで記述する、接続を確立することと、
前記リドゥログレコードのリプレイなしで、一つ以上のアクセス要求のための前記データベースへのアクセスを提供することと、
を実施するデータベースシステムのデータベースヘッドノードを実装するプログラム命令を記憶する、非一時的なコンピュータ可読記憶媒体。
【請求項１４】
前記複数のストレージノードで既に受信された前記複数のリドゥログレコードは、前記データベースヘッドノードとは異なるデータベースヘッドノードから受信された、請求項１３に記載の非一時的なコンピュータ可読記憶媒体。
【請求項１５】
前記データベースヘッドノードは、
前記データベースのためのアクセス要求を受信することと、
前記アクセス要求を受信することに応えて、前記一つ以上のストレージノードから前記データベースのための前記データの部分を記憶する一つ以上のデータページの現在の状態を要求することと、
前記アクセス要求にサービスを提供するために前記データベースのための前記データの部分を記憶する前記一つ以上のデータページの現在の状態を受信することであって、前記一つ以上の受信したデータページのうちの前記少なくとも一つの前記現在の状態は、前記一つ以上のストレージノードのうちの一つが、前記少なくとも一つのデータページの既に記憶されたバージョンまで前記複数のリドゥログレコードのうちの一つ以上をリプレイすることによって生成されることと、
をさらに実施する、請求項１３に記載の非一時的なコンピュータ可読記憶媒体。
【発明の詳細な説明】
【技術分野】
【０００１】
ソフトウェアスタックの多様な構成要素の分散は、いくつかの場合、（例えば複製によって）フォルトトレランス、より高い耐久性、及び（例えば、より少ない大型の高価な構成要素よりむしろ、多くのより小型でより安価な構成要素を使用することにより）より安価な解決策を提供する（または支援する）ことができる。ただし、データベースは、従来、分散の影響を最も受けにくいソフトウェアスタックの構成要素の中にある。例えば、データベースが提供すると期待されているいわゆるＡＣＩＤプロパティ（例えば、原子性、一貫性、独立性、及び永続性）を保証しつつもデータベースを分散することは困難であることがある。

【背景技術】
【０００２】
大部分の既存のリレーショナルデータベースは分散化されていないが、いくつかの既存のデータベースは、２つの共通モデル、つまり「シェアードナッシング」モデル及び「シェアードディスク」モデルの内の１つを使用して（より大型のモノリシックシステムを単に利用することによって「スケールアップ」されることと対照的に）「スケールアウト」される。一般的に、「シェアードナッシング」モデルでは、受信されたクエリーは（それぞれがクエリーの構成要素を含む）データベースシャードに分解され、これらのシャードはクエリー処理のために異なる計算ノードに送られ、結果は、結果が返される前に収集され、統合される。一般的に「シェアードディスク」モデルでは、クラスタのあらゆる計算ノードは同じ基礎的データにアクセスできる。このモデルを利用するシステムでは、キャッシュコヒーレンシーを管理するために細心の注意を払う必要がある。これらのモデルの両方において、大型のモノリシックデータベースは（スタンドアロンデータベースインスタンスの機能性のすべてを含んだ）複数のノードで複製され、それらを縫い合わせるために「グルー」ロジックが追加される。例えば、「シェアードナッシング」モデルでは、グルーロジックは、クエリーを再分割し、クエリーを複数の計算ノードに送信し、次いで結果を結合するディスパッチャーの機能性を提供してよい。「シェアードディスク」モデルでは、グルーロジックが（例えば、キャッシング層でコヒーレンシーを管理するために）複数のノードのキャッシュをともに融合させるのに役立ってよい。これらの「シェアードナッシング」データベースシステム及び「シェアードディスク」データベースシステムは配備するのが高価であり、維持するのが複雑であり、多くのデータベース使用ケースにサービスを提供しすぎる（ｏｖｅｒ－ｓｅｒｖｅ）ことがある。
【０００３】
実施形態は、いくつかの実施形態及び例示的な図面について一例として本明細書に説明されているが、当業者は実施形態が説明されている実施形態または図面に制限されないことを認識する。図面及び図面に対する詳細な説明は、開示されている特定の形式に実施形態を制限することを目的とするのではなく、逆に、添付の特許請求の範囲によって定められる精神及び範囲に入るすべての修正形態、同等物、及び変更形態を対象とすることを目的とすることが理解されるべきである。本明細書に使用される見出しは編成のためだけであり、明細書または特許請求項の範囲を制限するために使用されることを意図していない。本願を通して使用されるように、単語「してよい」は、強制の意味（つまり、しなければならないを意味する）よりむしろ、許可の意味（つまり、する可能性を有することを意味する）で使用される。単語「含む」、「含んだ」、及び「含む」はオープンエンド関係を示し、したがって含むが、これに限定されるものではないことを意味する。同様に、単語「有する」、「有している」、及び「有する」もオープンエンド関係を示し、したがって有するが、これに限定されるものではないことを意味する。本明細書で使用される用語「第１の」、「第２の」、「第３の」等は、それらが前に来る名詞に対するラベルとして使用され、いかなるタイプの順序付け（例えば、空間的、時間的、論理的等）も、係る順序付けがはっきりと特記されない限り暗示しない。
【０００４】
多様な構成要素は、１つまたは複数のタスクを実行する「ように構成される」として記述されてよい。係る文脈では、「ように構成される」は、動作中に１つまたは複数のタスクを実行する「構造を有する」を概して意味する大まかな記述である。したがって、構成要素は、構成要素が現在そのタスクを実行していなくてもタスクを実行するように構成できる（例えば、コンピュータシステムは、動作が現在実行されていなくても動作を実行するように構成されてよい）。いくつかの文脈では、「ように構成される」は、動作中に１つまたは複数のタスクを実行する「回路網を有する」を概して意味する構造の大まかな記述であってよい。したがって、構成要素は、構成要素が現在オンでなくてもタスクを実行するように構成できる。一般的に、「ように構成される」に対応する構造を形成する回路網はハードウェア回路を含んでよい。

【０００５】
多様な構成要素は、説明での便宜上、１つまたは複数のタスクを実行すると記述されてよい。係る説明は、言い回し「ように構成される」を含んでいるとして解釈されるべきである。１つまたは複数のタスクを実行するように構成される構成要素を記述することは、その構成要素に対して特許法第１１２条、第６項の解釈を行使することを明白に目的としていない。
【０００６】
「に基づいて」。本明細書に使用されるように、この用語は、決定に影響を及ぼす１つまたは複数の要因を説明するために使用される。この用語は、決定に影響を及ぼすことがある追加の要因を除外しない。すなわち、決定は、それらの要因だけに基づいてよい、または少なくとも部分的にそれらの要因に基づいてよい。言い回し「Ｂに基づいて Ａを決定する」を考える。ＢがＡの決定に影響を及ぼす要因であることがある一方、係る言い回しは、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だけに基づいて決定されてよい。
【０００７】
本開示の範囲は、本明細書に（明示的または暗示的のどちらかで）開示される任意の特徴または特徴の組合せまたはその任意の一般論を、それが本明細書で扱われる課題のいずれかまたはすべてを軽減するか否かに関わらず含む。したがって、特徴の係る任意の組合せに対して、本願（または本願に対する優先権を主張する出願）の手続き処理中に新しい特許請求の範囲が策定されることがある。特に、添付特許請求の範囲に関して、従属請求項からの特徴は独立請求項の特徴と組み合されてよく、それぞれの独立請求項からの特徴は任意の適切な方法で、及び単に添付の特許請求の範囲に列挙される特定の組合せでではなく、組み合されてよい。
【図面の簡単な説明】
【０００８】
【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多様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従って、ウェブ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てよい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、データベースエンジン、及び別個の分散型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多様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、分散型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、データベースシステムでの別個の分散型データベース最適化ストレージシステムの使用を示すブロック図である。
【図６】一実施形態に係る、分散型データベース最適化ストレージシステムの所与のノードにデータ及びメタデータがどのように記憶されてよいのかを示すブロック図である。
【図７】一実施形態に係る、データベースボリュームの例の構成を示すブロック図である。
【図８】いくつかの実施形態に係る、分散型データベースシステムでのシステム全体のチェックポイント回避のための方法を示す流れ図である。
【図９Ａ】いくつかの実施形態に係る、分散型データベースシステムのための高速クラッシュ回復を実行する方法を明示する一連の図である。
【図９Ｂ】いくつかの実施形態に係る、分散型データベースシステムのための高速クラッシュ回復を実行する方法を示す流れ図である。
【図９Ｃ】いくつかの実施形態に係る、回復されたデータベースでアクセス要求を処理するための方法を示す流れ図である。
【図１０】多様な実施形態に従って、データベースエンジン、及び別個の分散型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少なくとも一部を実装するように

構成されるコンピュータシステムを示すブロック図である。
【発明を実施するための形態】
【０００９】
分散型データベースシステムのためのシステム全体のチェックポイント回避の多様な実施形態が開示される。分散型ストレージシステムのストレージノードは、いくつかの実施形態では、ストレージノードに記憶される特定のデータページにリンクされる１つまたは複数のリドゥログレコードをデータベースシステムから受信してよい。データページは、データベースのためのデータを記憶する複数のデータページの内の１つであってよい。合体イベントは、特定のデータページにリンクされる１つまたは複数のリドゥログレコードに少なくとも部分的に基づいて特定のデータページについて検出されてよい。合体動作は、特定のデータページの以前に記憶されていたバージョンに１つまたは複数のログレコードを適用して、特定のデータページをその現在の状態で生成するために実行されてよい。
【００１０】
分散型データベースシステムのための高速クラッシュ回復の多様な実施形態が開示される。データベースシステムヘッドノードは、いくつかの実施形態では、故障回復動作を実行してよい。システム故障から回復すると、データベースのためのデータを記憶する分散型ストレージシステムのストレージノードとの接続が確立されてよい。いくつかの実施形態では、ストレージノードとの接続の確立時、データベースヘッドノードは、データベースをアクセスのために利用できるようにしてよい。少なくともいくつかの実施形態では、１つまたは複数のアクセス要求が受信されてよく、１つまたは複数のデータページの現在の状態が要求され、ストレージノードから受信されてよい。
【００１１】
明細書は、まず、システム全体のチェックポイント回避（例えば、作成、削除、使用、操作等）及び高速クラッシュ回復の技法を実装するように構成される例のウェブサービスベースのデータベースサービスを説明する。例のウェブサービスベースのデータベースサービスの説明に含まれているのは、データベースエンジン及び別個の分散型データベースストレージサービス等の、例のウェブサービスベースのデータベースサービスの多様な態様である。明細書は、次いでシステム全体のチェックポイント回避及び高速クラッシュ回復のための方法の多様な実施形態のフローチャートを説明する。次に、明細書は、開示されている技法を実装してよい例のシステムを説明する。明細書を通して多様な例が提供される。
【００１２】
本明細書に説明されるシステムは、いくつかの実施形態では、クライアント（例えば、加入者）がクラウドコンピューティング環境でデータストレージシステムを操作できるようにするウェブサービスを実装してよい。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且つ拡張可能である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、クエリーは複数の物理リソース全体で分散されるデータベースストレージに向けられてよく、データベースシステムは必要に応じてスケールアップ、またはスケールダウンされてよい。データベースシステムは、異なる実施形態で、多様なタイプ及び／または編成のデータベーススキーマと効果的に機能してよい。いくつかの実施形態では、クライアント／加入者は、例えばＳＱＬインタフェースを介してデータベースシステムに対話的に等、いくつかの方法でクエリーを提出してよい。他の実施形態では、外部アプリケーション及びプログラムは、データベースシステムにオープンデータベースコネクティビティ（ＯＤＢＣ）ドライバインタフェース及び／またはＪａｖａデータベースコネクティビティ（ＪＤＢＣ）ドライバインタフェースを使用してクエリーを提出してよい。
【００１３】
すなわち、本明細書に説明されるシステムは、いくつかの実施形態では、単一のデータベースシステムの多様な機能構成要素が本質的に分散されるサービス指向型データベースアーキテクチャを実装してよい。これらのシステムは、例えば、（それぞれが、アプリケ

ーションサーバ、サーチ機能性、またはデータベースのコア機能を提供するために必要とされる機能性を超える他の機能性等の外来の機能性を含んでよい）複数の完全でモノリシックなデータベースインスタンスを束ねるよりむしろ、データベースの基本的な動作（例えば、クエリー処理、トランザクション管理、キャッシング、及び記憶）を、個々に且つ無関係にスケーラブルであってよい階層に編成してよい。例えば、いくつかの実施形態では、本明細書に説明されるシステムの各データベースインスタンスは、（単一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んでよい）データベース階層、及び（既存のシステムのデータベース階層で従来実行される動作のいくつかを集合的に実行する複数のストレージノードを含んでよい）別個の分散されたストレージシステムを含んでよい。

【００１４】

本明細書により詳細に説明されるように、いくつかの実施形態では、データベースの最低レベルの動作（例えば、バックアップ動作、復元動作、スナップショット動作、回復動作、ログレコード操作動作、及び／または多様なスペース管理動作）のいくつかは、データベースエンジンからストレージ層にオフロードされ、複数のノード及びストレージデバイス全体で分散されてよい。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（またはデータベースのデータページ）に変更を適用し、次いで修正されたデータページをストレージ層に送信するよりむしろ、記憶されているデータベース（及びデータベースのデータページ）に対する変更の適用は、ストレージ層自体の責任であってよい。係る実施形態では、修正されたデータページよりむしろ、リドゥログレコードがストレージ層に送信されてよく、その後リドゥ処理（例えば、リドゥログレコードの適用）はいくぶんゆったりと且つ（例えば、バックグラウンドプロセスによって等）分散された方法で実行されてよい。いくつかの実施形態では、クラッシュ回復（例えば、記憶されているリドゥログレコードからのデータページの再構築）は、ストレージ層によって実行されてもよく、分散された（及び、いくつかの場合、ゆったりとした）バックグラウンドプロセスによって実行されてもよい。

【００１５】

いくつかの実施形態では、リドゥログだけ（及び修正されたデータページではない）がストレージ層に送信されるため、データベース階層とストレージ層との間にあるネットワークトラフィックは、既存のデータベースシステムにおいてよりもはるかに少なくてよい。いくつかの実施形態では、各リドゥログは、各リドゥログが変更を指定する対応するデータページのサイズのほぼ１０分の１であってよい。データベース階層及び分散型ストレージシステムから送信される要求が非同期であってよいこと、及び複数の係る要求が一度に送信中であってよいことに留意されたい。

【００１６】

一般的に、１個のデータを与えられた後、データベースの主要な要件は、最終的のその１個のデータを返すことができることである。これを行うために、データベースはそれぞれが異なる機能を実行するいくつかの異なる構成要素（または階層）含んでよい。例えば、従来のデータベースは３つの階層、つまりクエリーパーシング、最適化、及び実行を実行するための第１の階層、トランザクション性（ｔｒａｎｓａｃｔｉｏｎａｌｉｔｙ）、回復、及び耐久性を提供するための第２の階層、及びローカルでアタッチされたディスクでまたはネットワークでアタッチされたストレージのどちらかでストレージを提供する第３の階層を有すると見なされてよい。上述されたように、従来のデータベースをスケーリングしようとする以前の試みは、通常、データベースの３つすべての階層を複製し、そ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全体で分散することを伴っていた。

【００１７】

いくつかの実施形態では、本明細書に説明されるシステムは、従来のデータベースにおいてとは異なってデータベースシステムの機能性を仕切ってよく、スケーリングを実装するために複数のマシン全体で（完全なデータベースインスタンスよりもむしろ）機能構成

要素のサブセットだけを分散してよい。例えば、いくつかの実施形態では、クライアントが面する階層は、どのデータが記憶されるべきなのか、または取り出されるべきなのかを指定するが、どのようにしてデータを記憶するのか、または取り出すのかは指定しない要求を受信するように構成されてよい。この階層は、要求のパーシング及び／または最適化（例えば、ＳＱＬのパーシング及び要求）を実行してよい。一方、別の階層が、クエリーの実行に責任を負ってよい。いくつかの実施形態では、第３の階層が結果のトランザクション性及び一貫性を提供することに責任を負ってよい。例えば、この階層は、いわゆるＡＣＩＤプロパティのいくらか、特にデータベースをターゲットとするトランザクションの原子性を強化するよう構成されてよく、データベースの中で一貫性を維持し、データベースをターゲットとするトランザクション間で独立性を保証する。いくつかの実施形態では、第４の階層が次いで多様な種類の障害が存在する場合に記憶されているデータの耐久性を提供することに責任を負ってよい。例えば、この階層は、ロギングの変更、データベースクラッシュからの回復、基礎的な記憶ボリュームに対するアクセスの管理、及び／または基礎的な記憶ボリュームにおけるスペース管理に責任を負ってよい。

【００１８】

ここで図を参照すると、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多様な構成要素を示すブロック図である。この例に示されるように、データベースインスタンスは、それぞれがデータベースインスタンスの機能性の一部を提供する、複数の機能構成要素（または層）を含んでよい。この例では、データベースインスタンス１００は、（１１０として示される）クエリーパーシング及びクエリー最適化層、（１２０として示される）クエリー実行層、（１３０として示される）トランザクション性及び一貫性管理層、並びに（１４０として示される）耐久性及びスペース管理層を含む。上述されたように、いくつかの既存のデータベースシステムでは、データベースインスタンスのスケーリングは、（図１に示される層のすべてを含んだ）データベースインスタンス全体を１回または複数回複製して、次いで層を互いに縫い合わせるためにグルーロジックを追加することを含んでよい。いくつかの実施形態では、本明細書に説明されるシステムは、代わりにデータベース階層から別個のストレージ層に耐久性及びスペース管理層１４０の機能性をオフロードしてよく、その機能性をストレージ層の複数のストレージノード全体で分散してよい。

【００１９】

いくつかの実施形態では、本明細書に説明されるデータベースシステムは、図１に示されるデータベースインスタンスの上半分の構造の多くを保持してよいが、バックアップ動作、復元動作、スナップショット動作、回復動作、及び／または多様なスペース管理動作の少なくとも部分に対する責任を記憶階層に再配分してよい。このようにして機能性を再配分し、データベース階層と記憶階層との間でログ処理をしっかりと結合することは、スケーラブルデータベースを提供する以前の手法と比較されるときに性能を改善し、可用性を高め、コストを削減してよい。例えば、（実際のデータページよりもサイズがはるかに小さい）リドゥログレコードだけがノード全体で送り出される、または書込み動作のレーテンシパスの中で持続してよいので、ネットワーク及び入出力帯域幅の要件が削減されてよい。さらに、データページの生成は、入信書込み動作を遮ることなく、（フォアグラウンド処理が許すので）各ストレージノードでバックグラウンドで独立して実行できる。いくつかの実施形態では、ログ構造化された非上書きストレージの使用が、例えばデータページの移動またはコピーよりむしろメタデータ操作を使用することによって、バックアップ動作、復元動作、スナップショット動作、ポイントインタイムリカバリ動作、及びボリューム増大動作をより効率的に実行できるようにしてよい。いくつかの実施形態では、ストレージ層は、複数のストレージノード全体でクライアントの代わりに記憶されたデータの複製（及び／またはリドゥログレコード等の、そのデータと関連付けられたメタデータ）に対する責任を負ってもよい。例えば、データ（及び／またはメタデータ）は、（例えば、ストレージノードの集合体が独自の物理的に別個の独立したインフラストラクチャで実行する単一の「可用性ゾーン」の中で等）ローカルに、及び／または単一の領域のもし

くは異なる領域の可用性ゾーン全体で複製されてよい。

【００２０】

多様な実施形態では、本明細書に説明されるデータベースシステムは、さまざまなデータベース動作のために標準的な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フェース（ＡＰＩ）をサポートしてよい。例えば、ＡＰＩは、データベースの作成、テーブルの作成、テーブルの改変、ユーザーの作成、ユーザーの削除、テーブルでの１行または複数行の挿入、値のコピー、テーブルの中からのデータの選択（例えば、テーブルの問合せ）、クエリーの取消しまたはアボート、スナップショットの作成のための動作、及び／または他の動作をサポートしてよい。

【００２１】

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、多様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）からの読取り要求及び／または書込み要求を受信し、次いで要求をパースし、関連付けられたデータベース動作（複数の場合がある）実施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を含んでよい。例えば、データベースエンジンヘッドノードは、複雑なクエリー及び接合の結果を得るために必要な一連のステップを作成してよい。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型データベース最適化ストレージシステムとの間の通信だけではなく、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も管理してよい。

【００２２】

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、ＪＤＢＣインタフェースまたはＯＤＢＣインタフェースを通してエンドクライアントからＳＱＬ要求を受信すること、及びローカルでＳＱＬ処理及び（ロッキングを含んでよい）トランザクション管理を実行することに責任を負ってよい。ただし、データベースエンジンヘッドノード（またはデータベースエンジンヘッドノードの多様な構成要素）は、データページをローカルで生成するよりむしろ、リドゥログレコードを生成してよく、リドゥログレコードを別個の分散型ストレージシステムの適切なノードに送り出してよい。いくつかの実施形態では、分散型ストレージシステムのためのクライアント側ドライバは、データベースエンジンヘッドノードでホストされてよく、それらのリドゥログレコードが向けられるセグメント（またはセグメントのデータページ）を記憶する１つのストレージシステムノード（または複数のストレージシステムノード）にリドゥログレコードを送ることに責任を負ってよい。例えば、いくつかの実施形態では、各セグメントは保護グループを形成する複数のストレージシステムノードでミラーリングされてよい（またはそれ以外の場合、耐久的にされてよい）。係る実施形態では、クライアント側ドライバは、各セグメントが記憶されるノードを追跡調査してよく、クライアント要求が受信されるときに（例えば非同期で、及び実質的にほぼ同時に並列で）セグメントが記憶されるノードのすべてにリドゥログを送ってよい。クライアント側ドライバが（リドゥログレコードがストレージノードに書き込まれていることを示すことがある）保護グループのストレージノードの書込み選抜グループ（ｑｕｏｒｕｍ）から肯定応答を受信するとすぐに、クライアント側ドライバはデータベース階層に（例えば、データベースエンジンヘッドノードに）要求された変更の肯定応答を送信してよい。例えば、データが保護グループを使用することによって耐久的にされる実施形態では、データベースエンジンヘッドノードは、クライアント側ドライバが書込み選抜グループを構成するために十分なストレージノードインスタンスから回答を受信するまで及び受信しない限り、トランザクションをコミットできないことがある。同様に、特定のセグメントに向けられる読取り要求の場合、クライアント側ドライバは、（例えば非同期で、及び実質的に同時に並列で）セグメントが記憶されるノードのすべてに読取り要求を送ってよい。クライアント側ドライバは保護グループのストレージノードの読取り選抜グループから要求されたデータを受信するとすぐに、クライアント側ドライバはデータベース階層に（例えば、データベースエンジンヘッドノードに）要求されたデータを返

してよい。

【００２３】

いくつかの実施形態では、データベース階層（またはより詳細には、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が一時的に保持されるキャッシュを含んでよい。係る実施形態では、係るキャッシュに保持されるデータページをターゲットとする書込み要求が受信されると、対応するリドゥログレコードをストレージ層に送り出すことに加えて、データベースエンジンはそのキャッシュに保持されているデータページのコピーに変更を適用してよい。ただし、他のデータベースシステムにおいてとは異なり、このキャッシュに保持されるデータページはストレージ層にフラッシュされることはなく、該データページはいつでも（例えば、キャッシュに入れられたコピーに最も最近に適用された書込み要求のリドゥログレコードがストレージ層に送信され、肯定応答された後のいつでも）廃棄されてよい。キャッシュは、異なる実施形態で、一度に多くても一人の書込み者（または複数の読取り者）によるキャッシュへのアクセスを制御するための多様なロッキング機構のいずれかを実装してよい。ただし、係るキャッシュを含む実施形態では、キャッシュは複数のノード全体で分散れるのではなく、所与のデータベースインスタンスのためにデータベースエンジンヘッドノードだけに存在してよいことに留意されたい。したがって、管理するキャッシュコヒーレンシーまたは一貫性問題がないことがある。

【００２４】

いくつかの実施形態では、データベース階層は、例えば、読取り要求を送ることができるデータベース階層の異なるノードでのデータの読取り専用コピー等、システムでの同期または非同期の読取りレプリカの使用をサポートしてよい。係る実施形態では、所与のデータベースのデータベースエンジンヘッドノードが特定のデータページに向けられる読取り要求を受信すると、データベースエンジンヘッドノードはこれらの読取り専用コピーの内のいずれか１つ（または特定の１つ）に要求を送ってよい。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のクライアント側ドライバは、（例えば、これらの他のノードにそのキャッシュを無効にするように促すために）キャッシュに入れられたデータページに対する更新及び／または失効についてこれらの他のノードに通知するように構成されてよい（その後これらの他のノードはストレージ層から更新されたデータページの更新済みのコピーを要求してよい）。

【００２５】

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で実行中のクライアント側ドライバは、記憶階層にプライベートインタフェースを曝露してよい。いくつかの実施形態では、クライアント側ドライバは従来のｉＳＣＳＩインタフェースを１つまたは複数の他の構成要素（例えば、他のデータベースエンジンまたは仮想コンピューティングサービス構成要素）に曝露してもよい。いくつかの実施形態では、記憶階層でのデータベースインスタンスのためのストレージは、制限なくサイズを増大することがあり、それと関連付けられた、制限されない数のＩＯＰＳを有することがある単一のボリュームとしてモデル化されてよい。ボリュームが作成されるとき、ボリュームは特定のサイズで、（例えば、ボリュームがどのように複製されるのかを指定する）特定の可用性／耐久性特徴で、及び／またはボリュームと関連付けられたＩＯＰＳレートで（例えば、ピークと持続の両方）作成されてよい。例えば、いくつかの実施形態では、さまざまな異なる耐久性モデルがサポートされてよく、ユーザー／加入者は自らのデータベースのために、複製コピー、ゾーン、もしくは領域の数、及び／またはその耐久性、性能、及びコストの目的に基づいて複製が同期であるのか、それとも非同期であるのかを指定できてよい。

【００２６】

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ついてのメタデータを維持してよく、ストレージノード間で追加のホップを必要とすることなく、読取り要求及び書込み要求を実行するため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的に送信してよい。例えば、いくつかの実施形態で、データベースに対する変更を行う要

求に応えて、クライアント側ドライバは、ターゲットとされたデータページのストレージを実装している１つまたは複数のノードを決定し、それらのストレージノードに対してその変更を指定するリドゥログレコード（複数の場合がある）を送るように構成されてよい。ストレージノードは、次いで、リドゥログレコードに指定される変更を将来のある時点でターゲットとされたデータページに適用することに責任を負ってよい。書込みはクライアント側ドライバに肯定応答されるので、クライアント側ドライバは、ボリュームが耐久的となる点を先に進めてよく、データベース階層に対してコミットを肯定応答してよい。上述されたように、いくつかの実施形態では、クライアント側ドライバはストレージノードサーバにデータページを絶対に送信しないことがある。これは、ネットワークトラフィックを削減するだけではなく、チェックポイントまたは以前のデータベースシステムでのフォアグラウンド処理スループットを制約するバックグラウンド書込み者スレッドの必要性を削除してもよい。

【００２７】

いくつかの実施形態では、多くの読取り要求がデータベースエンジンヘッドノードキャッシュによって提供されてよい。ただし、大規模故障イベントは一般的すぎて、メモリ内複製だけを許可できないので、書込み要求は耐久性を必要としてよい。したがって、本明細書に説明されるシステムは、記憶階層内のデータストレージを２つの領域、つまりリドゥログレコードがデータベース階層から受信されるときにリドゥログレコードが書き込まれる小さなアペンド専用ログ構造化領域、及びバックグラウンドでデータページの新しいバージョンを作成するために、ログレコードがともに合体するより大きな領域として実装することによって、フォアグラウンドレーテンシパス内にあるリドゥログレコード書込み動作のコストを最小限に抑えるように構成されてよい。いくつかの実施形態では、メモリ内構造は、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連鎖ログレコード後方へ、データページの前回のリドゥログレコードを指すデータページごとに維持される。この手法は、読取りがおもにキャッシュに入れられるアプリケーション内を含んで、混合した読取り‐書込みワークロードに優れた性能を提供してよい。

【００２８】

いくつかの実施形態では、リドゥログレコードのためのログ構造化データストレージへのアクセスは、（ランダム入出力動作よりむしろ）一連の順次入出力動作から構成されてよいため、行われている変更は互いに密接にパックされてよい。データページに変更するたびに、永続データストレージに対する２つの入出力動作（リドゥログのための動作及び修正されたデータページ自体のための動作）が生じる既存のシステムとは対照的に、いくつかの実施形態では、本明細書に説明されるシステムはリドゥログレコードの受信に基づいて分散型ストレージシステムのストレージノードでデータページを合体させることによってこの「書込み増幅」を回避してよい。

【００２９】

上述されたように、いくつかの実施形態では、データベースシステムの記憶階層はデータベーススナップショットを撮ることに責任を負ってよい。ただし、記憶階層はログ構造化ストレージを実装するため、データページ（例えば、データブロック）のスナップショットを撮ることはデータページ／ブロックに最も最近適用されたリドゥログレコードと関連付けられたタイムスタンプ（またはデータページ／ブロックの新しいバージョンを作成するために複数のリドゥログレコードを合体させるための最も最近の動作と関連付けられたタイムスタンプ）を記録すること、及びページ／ブロックの以前のバージョン及び時間内に記録された点までのあらゆる以後のログエントリのガベージコレクションを妨げることを含んでよい。係る実施形態では、データベーススナップショットを撮ることは、オフボリュームバックアップ戦略を利用するときに必要とされるだろう、データブロックの読取り、コピー、または書込みを必要としないことがある。いくつかの実施形態では、ユーザー／加入者はアクティブデータセットに加えてオンボリュームスナップショットのためにどれほど多くの追加スペースを保つことを希望するのかを選ぶことができることがあるが、修正されたデータだけが追加のスペースを必要とするので、スナップショットのスペ

ース要件は最小であってよい。異なる実施形態では、スナップショットは、不連続（例えば、各スナップショットは時間の特定の時点でのデータページ内のデータのすべてに対するアクセスを提供してよい）または連続（例えば、各スナップショットは２つの時点の間のデータページに存在するデータのすべてのバージョンに対するアクセスを提供してよい）であってよい。いくつかの実施形態では、以前のスナップショットに戻ることは、そのスナップショット以降のすべてのリドゥログレコード及びデータページが無効であり、ガベージコレクション可能であることを示すためにログレコードを記録すること、及びスナップショット点後のすべてのデータベースキャッシュエントリを廃棄することを含んでよい。係る実施形態では、ストレージシステムは、ストレージシステムが通常の順方向読取り／書込み処理で行うのと同様に、要求されるように、及びすべてのノード全体でバックグラウンドで、ブロック単位でリドゥログレコードをデータブロックに適用するので、前進復帰は必要とされないことがある。クラッシュ回復は、それによってノード全体で並列且つ分散型にされてよい。

【００３０】

ウェブ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てよいサービスシステムアーキテクチャの一実施形態が図２に示される。示されている実施形態では、（データベースクライアント２５０ａから２５０ｎとして示される）多くのクライアントがネットワーク２６０を介してウェブサービスプラットホーム２００と対話するように構成されてよい。ウェブサービスプラットホーム２００は、データベースサービス２１０、分散型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つまたは複数の他の仮想コンピューティングサービス２３０の１つまたは複数のインスタンスとインタフェースをとるように構成されてよい。所与の構成要素の１つまたは複数が存在してよい場合、本明細書でのその構成要素に対する参照は単数形または複数形のどちらかで行われてよいことが留意される。ただしどちらの形の使用も他方を排除することを目的としていない。

【００３１】

多様な実施形態では、図２に示される構成要素は、コンピュータハードウェア（例えば、マイクロプロセッサもしくはコンピュータシステム）によって直接的にまたは間接的に実行可能な命令として、またはこれらの技法の組合せを使用してコンピュータハードウェアの中で直接的に実装されてよい。例えば、図２の構成要素はそれぞれが図１０に示され、以下に説明されるコンピュータシステム実施形態に類似してよい、いくつかのコンピューティングノード（つまり、単にノード）を含むシステムによって実装されてよい。多様な実装形態では、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）の機能性は、特定のノードによって実装されてよい、またはいくつかのノード全体で分散されてよい。いくつかの実施形態では、所与のノードは複数のサービスシステム構成要素（例えば、複数のデータベースサービスシステム構成要素）の機能性を実装してよい。

【００３２】

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に対する要求（例えば、スナップを生成する要求等）を含むウェブサービス要求を、ネットワーク２６０を介してウェブサービスプラットホーム２００に提出するように構成可能な任意のタイプのクライアントを包含してよい。例えば、所与のクライアント２５０は、ウェブブラウザの適切なバージョンを含んでよい、またはウェブブラウザによって提供される実行環境に対する拡張部として、またはウェブブラウザによって提供される実行環境の中で実行するように構成されるプラグインモジュールまたは他のタイプのコードモジュールを含んでよい。代わりに、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、データベースアプリケーション（もしくはデータベースアプリケーションのユーザーインタフェース）、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１つまたは複数のデータベースを記憶する、及び／または１つまたは複数のデータベースにアクセスするために永続記憶装置リソースを利用してよい任意の他のアプリケーション等のアプリケー

ションを包含してよい。いくつかの実施形態では、係るアプリケーションは、必ずしもすべてのタイプのウェブベースのデータに対する完全なブラウザサポートを実装しなくてもウェブサービス要求を生成し、処理するための（例えば、ハイパテキスト転送プロトコル（ＨＴＴＰ）の適切なバージョンのための）十分なプロトコルサポートを含んでよい。すなわち、クライアント２５０は、ウェブサービスプラットホーム２００と直接的に対話するように構成される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、表象状態転送（Ｒｅｐｒｅｓｅｎｔａｔｉｏｎａｌ Ｓｔａｔｅ Ｔｒａｎｓｆｅｒ）（ＲＥＳＴ）様式ウェブサービスアーキテクチャ、ドキュメントベースもしくはメッセージベースのウェブサービスアーキテクチャ、または別の適切なウェブサービスアーキテクチャに従ってウェブサービス要求を生成するよう構成されてよい。

【００３３】

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、データベースのウェブサービスベースのストレージへのアクセスを、他のアプリケーションに、それらのアプリケーションにはトランスペアレントな方法で提供するように構成されてよい。例えば、クライアント２５０は、オペレーティングシステムまたはファイルシステムと統合して、本明細書に説明されるストレージモデルの適切な変形に従ってストレージを提供するように構成されてよい。ただし、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／またはフォルダの従来のファイルシステム階層等の、アプリケーションに異なるストレージインタフェースを提示してよい。係る実施形態では、アプリケーションは図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修正される必要はないことがある。代わりに、ウェブサービスプラットホーム２００へのインタフェースをとることの詳細は、オペレーティングシステム環境の中で実行するアプリケーションの代わりに、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てよい。

【００３４】

クライアント２５０は、ネットワーク２６０を介してウェブサービスプラットホーム２００にウェブサービス要求（例えば、スナップショット要求、スナップショット要求のパラメータ、読取り要求、スナップショットの復元等）を伝達し、ウェブサービスプラットホーム２００から応答を受信してよい。多様な実施形態では、ネットワーク２６０は、クライアント２５０とプラットホーム２００との間でウェブベースの通信を確立するために必要なネットワーキングハードウェア及びプロトコルの任意の適切な組合せを包含してよい。例えば、ネットワーク２６０は、集合的にインターネットを実装する多様な電気通信ネットワーク及びサービスプロバイダを概して包含してよい。また、ネットワーク２６０は、公衆無線ネットワークまたは構内無線ネットワークだけではなく、ローカルエリアネットワーク（ＬＡＮ）または広域ネットワーク（ＷＡＮ）等の構内ネットワークも含んでよい。例えば、所与のクライアント２５０とウェブサービスプラットホーム２００の両方とも、独自の内部ネットワークを有する企業の中でそれぞれプロビジョニングされてよい。係る実施形態では、ネットワーク２６０は、インターネットとウェブサービスプラットホーム２００との間だけではなく、所与のクライアント２５０とインターネットとの間にネットワーキングリンクを確立するために必要なハードウェア（例えば、モデム、ルータ、開閉器、ロードバランサ、プロキシサーバ等）及びソフトウェア（例えば、プロトコルスタック、財務会計ソフト、ファイアウォール／セキュリティソフトウェア等）を含んでよい。いくつかの実施形態では、クライアント２５０は、公衆インターネットよりむしろ構内ネットワークを使用してウェブサービスプラットホーム２００と通信してよい。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／または分散型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）と同じ企業の中でプロビジョニングされてよい。係る場合、クライアント２５０は、構内ネットワーク２６０（例えば、インターネットベースの通信プロトコルを使用してよいが、公にアクセス可能ではないＬＡＮまたはＷＡＮ）を通して完全にプラットホーム２００と通信してよい。

【００３５】
一般的に言えば、ウェブサービスプラットホーム２００は、データページ（またはデータページのレコード）にアクセスする要求等のウェブサービス要求を受信し、処理するように構成される１つまたは複数のサービスエンドポイントを実装するように構成されてよい。例えば、ウェブサービスプラットホーム２００は、特定のエンドポイントを実装するように構成されるハードウェア及び／またはソフトウェアを含んでよく、したがってそのエンドポイントに向けられたＨＴＴＰベースのウェブサービス要求は適切に受信され、処理される。一実施形態では、ウェブサービスプラットホーム２００は、クライアント２５０からウェブサービス要求を受信し、ウェブサービス要求を、処理のためにデータベースサービス２１０、分散型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するように構成されるサーバシステムとして実装されてよい。他の実施形態では、ウェブサービスプラットホーム２００は、大規模なウェブサービス要求処理ロードを動的に管理するように構成されるロードバランス機能及び他の要求管理機能を実装する（例えば、クラスタトポロジの）いくつかの別個のシステムとして構成されてよい。多様な実施形態では、ウェブサービスプラットホーム２００は、ＲＥＳＴ様式またはドキュメントベースの（例えば、ＳＯＡＰベースの）タイプのウェブサービス要求をサポートするように構成されてよい。

【００３６】
いくつかの実施形態では、ウェブサービスプラットホーム２００は、クライアントのウェブサービス要求に対するアドレス可能なエンドポイントとして機能することに加えて、多様なクライアント管理機能を実装してよい。例えば、プラットホーム２００は、例えば要求側クライアント２５０のアイデンティティ、クライアント要求の数及び／または頻度、クライアント２５０の代わりに記憶されているまたは取り出されるデータテーブル（またはデータテーブルのレコード）のサイズ、クライアント２５０によって使用される全体的な記憶帯域幅、クライアント２５０によって要求されるストレージのクラス、または任意の他の測定可能なクライアント使用パラメータを追跡調査することによって、ストレージリソースを含むウェブサービスのクライアント使用の計量及びアカウンティングを調整してよい。プラットホーム２００は、財務会計システム及び請求書作成システムを実装してもよい、またはクライアント使用活動の報告及び請求書作成のために外部システムによって照会され、処理されてよい使用データのデータベースを維持してもよい。特定の実施形態では、プラットホーム２００は、クライアント２５０から受け取られる要求の割合及びタイプ、係る要求によって活用される帯域幅、係る要求のためのシステム処理レーテンシ、システム構成要素活用（例えば、ストレージサービスシステムの中のネットワーク帯域幅及び／またはストレージ活用）、要求から生じるエラーの割合及びタイプ、記憶され、要求されるデータページもしくはそのレコードの特徴（例えば、サイズ、データタイプ等）を反映する測定基準、または任意の他の適切な測定基準等、さまざまなストレージサービスシステム操作測定基準を収集する、監視する、及び／または統合するよう構成されてよい。いくつかの実施形態では、係る測定基準はシステム構成要素を調整し、維持するためにシステム管理者によって使用されてよい。一方、他の実施形態では、係る測定基準（または係る測定基準の関連性のある部分）は、係るクライアントがデータベースサービス２１０、分散型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基礎的なシステム）の使用を監視できるようにするためにクライアント２５０に曝露されてよい。

【００３７】
いくつかの実施形態では、プラットホーム２００は、ユーザー認証手順及びアクセス制御手順も実装してよい。例えば、特定のデータベースにアクセスする所与のウェブサービス要求の場合、プラットホーム２００は、要求と関連付けられるクライアント２５０が特定のデータベースにアクセスする権限を与えられているかどうかを確かめるように構成されてよい。プラットホーム２００は、例えばアイデンティティ、パスワード、もしくは他

の信用証明書を特定のデータベースと関連付けられた信用証明書に対して評価する、または特定のデータベースに対する要求されたアクセスを、特定のデータベースに対するアクセス制御リストに対して評価することによって係る権限付与を決定してよい。例えば、クライアント２５０が特定のデータベースにアクセスするほど十分な信用証明書を有していない場合、プラットホーム２００は、例えばエラー状態を示す応答を要求側クライアント２５０に返すことによって対応するウェブサービス要求を拒絶してよい。多様なアクセス制御方針は、データベースサービス２１０、分散型データベース最適化ストレージサービス２２０、及び／または他の仮想コンピューティングサービス２３０によってアクセス制御情報のレコードまたはリストとして記憶されてよい。

【００３８】

ウェブサービスプラットホーム２００が、クライアント２５０が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特徴にそれを通してアクセスしてよい一次インタフェースを表してよいが、ウェブサービスプラットホーム２００が係る特徴に対する単独のインタフェースを表す必要がないことが留意される。例えば、ウェブサービスインタフェースとは別個であってよい代替のＡＰＩは、データベースシステムを提供する企業にとって内部のクライアントがウェブサービスプラットホーム２００を迂回できるようにするために使用されてよい。本明細書に説明される例の多くで、分散型データベース最適化ストレージサービス２２０が、クライアント２５０にデータベースサービスを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にとって内部であってよく、外部クライアント（例えば、ユーザーまたはクライアントアプリケーション）に曝露されないことがあることに留意されたい。係る実施形態では、内部「クライアント」（例えば、データベースサービス２１０）は、（例えば、これらのサービスを実装するシステムの間で直接的にＡＰＩを通して）分散型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との間の実線として示されるローカルネットワークまたは構内ネットワーク上で分散型データベース最適化ストレージサービス２２０にアクセスしてよい。係る実施形態では、クライアント２５０の代わりにデータベースを記憶する上での分散型データベース最適化ストレージサービス２２０の使用はそれらのクライアントにとってトランスペアレントであってよい。他の実施形態では、分散型データベース最適化ストレージサービス２２０は、データベース管理のために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、データベースまたは他の情報のストレージを提供するために、ウェブサービスプラットホーム２００を通してクライアント２５０に曝露されてよい。これは、ウェブサービスプラットホーム２００と分散型データベース最適化ストレージサービス２２０の間の破線によって図２に示される。係る実施形態では、分散型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、ネットワーク２６０を介して（例えば、インターネット上で）分散型データベース最適化ストレージサービス２２０にアクセスしてよい。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、クライアント２５０の代わりにコンピューティングサービス２３０を実行する上で使用されるオブジェクトを記憶するために（例えば、仮想コンピューティングサービス２３０と分散型データベース最適化ストレージサービス２２０との間で直接的にＡＰＩを通して）分散型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受信するように構成されてよい。これは、仮想コンピューティングサービス２３０と分散型データベース最適化ストレージサービス２２０との間の破線によって図２に示される。いくつかのケースでは、プラットホーム２００のアカウンティングサービス及び／または信用証明書発行（ｃｒｅｄｅｎｔｉａｌｉｎｇ）サービスは、管理クライアント等の内部クライアントにとって、または同じ企業の中のサービス構成要素間では不必要となってよい。

【００３９】

多様な実施形態では、異なる記憶方針が、データベースサービス２１０及び／または分散型データベース最適化ストレージサービス２２０によって実装されてよいことに留意されたい。係る記憶方針の例は、耐久性方針（例えば、記憶されるデータベース（またはデ

ータベースのデータページ）のインスタンスの数、及びデータベースが記憶される異なるノードの数を示す方針）、及び／または（要求トラフィックを一様にしようとしてデータベースまたはデータベースのデータページを、異なるノード、ボリューム、及び／またはディスク全体で分散してよい）ロードバランシング方針を含んでよい。さらに、異なる記憶方針は、サービスの多様な１つによって異なるタイプの記憶された項目に適用されてよい。例えば、いくつかの実施形態では、分散型データベース最適化ストレージサービス２２０は、データページに対するよりもリドゥログレコードに対してより高い耐久性を実装してよい。

【００４０】

図３は、一実施形態に従って、データベースエンジン、及び別個の分散型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多様な構成要素を示すブロック図である。この例では、データベースシステム３００は、いくつかのデータベースのそれぞれのためのそれぞれのデータベースエンジンヘッドノード３２０、及び（データベースクライアント３５０ａから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可視であってよい、または可視でないことがある）分散型データベース最適化ストレージサービス３１０を含む。この例で示されるように、データベースクライアント３５０ａから３５０ｎの内の１つまたは複数は、データベースヘッドノード３２０（例えば、それぞれがそれぞれのデータベースインスタンスの構成要素である、ヘッドノード３２０ａ、ヘッドノード３２０ｂ、またはヘッドノード３２０ｃ）に、ネットワーク３６０を介してアクセスしてよい（例えば、これらの構成要素はネットワークアドレス指定可能且つデータベースクライアント３５０ａから３５０ｎにアクセス可能であってよい）。ただし、データベースクライアント３５０ａから３５０ｎの代わりに、１つまたは複数のデータベースのデータページ（及びリドゥレコード及び／またはそれと関連付けられた他のメタデータ）を記憶し、本明細書に説明されるようにデータベースシステムの他の機能を実行するためにデータベースシステムによって利用されてよい分散型データベース最適化ストレージサービス３１０は、異なる実施形態では、ネットワークアドレス指定可能且つストレージクライアント３５０ａから３５０ｎにアクセス可能であってよい、またはアクセス可能でないことがある。例えば、いくつかの実施形態では、分散型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から３５０ｎに非可視である方法で多様な記憶動作、アクセス動作、ロギング変更動作、回復動作、ログレコード操作動作、及び／またはスペース管理動作を実行してよい。

【００４１】

上述されたように、各データベースインスタンスは、多様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）から要求（例えば、スナップショット要求等）を受信し、次いで要求をパースし、要求を最適化し、関連付けられたデータベース動作（複数の場合がある）を実施するための実行計画を作成する単一のデータベースエンジンヘッドノード３２０を含んでよい。図３に示される例では、データベースエンジンヘッドノード３２０ａのクエリーパーシング、最適化、及び実行構成要素３０５は、データベースクライアント３５０ａから受信され、データベースエンジンヘッドノード３２０ａがその構成要素であるデータベースインスタンスをターゲットとするクエリーのためにこれらの機能を実行してよい。いくつかの実施形態では、クエリーパーシング、最適化、及び実行構成要素３０５はデータベースクライアント３５０ａに、書込み肯定応答、要求されたデータページ（及びデータページの部分）、エラーメッセージ、及びまたは他の応答を適宜に含んでよいクエリー応答を返してよい。この例に示されるように、データベースエンジンヘッドノード３２０ａは、分散型データベース最適化ストレージサービス３１０の中で多様なストレージノードに読取り要求及び／またはリドゥログレコードを送り、分散型データベース最適化ストレージサービス３１０から書込み肯定応答を受信し、分散型データベース最適化ストレージサービス３１０から要求されたデータページを受信し、及び／またはデータページ、エラーメッセージ、または他の応答を（同様にそれらをデータベースクライアント３５０ａに返してよい）クエリーパーシング、最適化、及

び実行構成要素３０５に返してよい、クライアント側ストレージサービスドライバ３２５も含んでよい。

【００４２】

この例では、データベースエンジンヘッドノード３２０ａ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てよいデータページキャッシュ３３５を含む。図３に示されるように、データベースエンジンヘッドノード３２０ａは、データベースエンジンヘッドノード３２０ａが構成要素であるデータベースインスタンスでトランザクション性及び一貫性を提供することに責任を負ってよいトランザクション及び一貫性管理構成要素３３０も含んでよい。例えば、この構成要素は、データベースインスタンス及び該データベースインスタンスに向けられるトランザクションの原子性、一貫性、及び独立性のプロパティを保証することに責任を負ってよい。図３に示されるように、データベースエンジンヘッドノード３２０ａは、多様なトランザクションのステータスを追跡調査し、コミットしないトランザクションのあらゆるローカルでキャッシュに入れられた結果をロールバックするためにトランザクション及び一貫性管理構成要素３３０によって利用されてよいトランザクションログ３４０及びアンドゥログ３４５も含んでよい。

【００４３】

図３に示される他のデータベースエンジンヘッドノード３２０（例えば、３２０ｂ及び３２０ｃ）のそれぞれが類似する構成要素を含んでよく、データベースクライアント３５０ａから３５０ｎの内の１つまたは複数によって受信され、それが構成要素であるそれぞれのデータベースインスタンスに向けられるクエリーのために類似する機能を実行してよいことに留意されたい。

【００４４】

いくつかの実施形態では、本明細書に説明される分散型データベース最適化ストレージシステムは、１つまたは複数のストレージノードでの記憶のために多様な論理ボリューム、セグメント、及びページでデータを編成してよい。例えば、いくつかの実施形態では、各データベースは論理ボリュームによって表され、各論理ボリュームはストレージノードの集合体上でセグメント化される。ストレージノード内の特定のストレージノード上で生きる各セグメントは、隣接ブロックアドレスのセットを含む。いくつかの実施形態では、各データページはセグメントに記憶され、したがって各セグメントは１つまたは複数のデータページの集合体及びそれが記憶する各データページの（リドゥログとも呼ばれる）変更ログ（例えば、リドゥログレコードのログ）を記憶する。本明細書に詳細に説明されるように、ストレージノードは（本明細書でＵＬＲとも呼ばれてよい）リドゥログレコードを受信し、リドゥログレコードを合体させて、（例えば、ゆったりと及び／またはデータページもしくはデータベースクラッシュに対する要求に応えて）対応するデータページ及び／または追加のもしくは代替のログレコードの新しいバージョンを作成するように構成されてよい。いくつかの実施形態では、データページ及び／または変更ログは（クライアントによって指定されてよく、クライアントの代わりにデータベースシステムでデータベースが維持されている）可変構成に従って複数のストレージノード全体でミラーリングされてよい。例えば、異なる実施形態では、データログまたは変更ログの１つのコピー、２つのコピー、または３つのコピーがデフォルト構成、アプリケーションに特有の耐久性優先度、またはクライアントによって指定される耐久性優先度に従って、１つ、２つ、または３つの異なる可用性ゾーンもしくは領域のそれぞれに記憶されてよい。

【００４５】

本明細書に使用されるように、以下の用語は、多様な実施形態に従って分散型データベース最適化ストレージシステムによってデータの編成を説明するために使用されてよい。

【００４６】

ボリューム：ボリュームは、ストレージシステムのユーザー／クライアント／アプリケーションが理解するストレージのきわめて耐久性のある単位を表す論理概念である。すなわち、ボリュームはデータベースの多様なユーザーページに対する書込み動作の単一の一貫性がある順序付けられたログとしてユーザー／クライアント／アプリケーションに見え

る分散型ストアである。各書込み動作は、ボリュームの中で単一のユーザーページのコンテンツに対する論理的な順序付けられた変形を表すユーザーログレコード（ＵＬＲ）で符号化されてよい。上述されたように、ＵＬＲは、本明細書でリドゥログレコードと呼ばれてもよい。各ＵＬＲは、一意の識別子（例えば、論理シーケンス番号（ＬＳＮ））を含んでよい。各ＵＬＲは、ＵＬＲに高い耐久性及び可用性を提供するために、保護グループ（ＰＧ）を形成する、分散型ストア内の１つまたは複数の同期セグメントに持続してよい。ボリュームは、バイトの可変サイズの連続範囲にＬＳＮ型の読取り／書込みインタフェースを提供してよい。

【００４７】

いくつかの実施形態では、ボリュームはそれぞれが保護グループを通して耐久的にされた複数のエクステントから構成されてよい。係る実施形態では、ボリュームはボリュームエクステントの変わりやすい連続シーケンスから構成されるストレージの単位を表してよい。ボリュームに向けられる読取り及び書込みは、構成するボリュームエクステントに対する対応する読取り及び書込みにマッピングされてよい。いくつかの実施形態では、ボリュームのサイズは、ボリュームエクステントを追加することにより、又は、ボリュームの端部からボリュームエクステントを除去することにより変更されてもよい。

【００４８】

セグメント：セグメントは、単一ストレージノードに割り当てられるストレージの制限される耐久性の単位である。すなわち、セグメントは、特有の固定サイズバイト範囲のデータに、限られたベストエフォート型の耐久性（例えば、ストレージノードである、故障の永続的であるが冗長ではない単一点）を提供する。多様な実施形態では、このデータは、いくつかの場合では、ユーザーアドレス指定可能なデータのミラーであってよい、またはこのデータはボリュームメタデータまたはイレイジャーコーディングされたビット等の他のデータであってよい。所与のセグメントは、正確に１つのストレージノード上で生きてよい。ストレージノードの中で、複数のセグメントが各ＳＳＤ上で生きてよく、各セグメントは１つのＳＳＤに制限されてよい（例えば、セグメントは複数のＳＳＤに及ばないことがある）。いくつかの実施形態では、セグメントはＳＳＤ上で連続領域を占有するように要求されないことがある。むしろ、各ＳＳＤにセグメントのそれぞれによって所有される領域を記述する割当てマップがあってよい。上述されたように、保護グループは複数のストレージノードに渡って拡散される複数のセグメントから構成されてよい。いくつかの実施形態では、セグメントは、（サイズが作成時に定義される）バイトの固定サイズの隣接範囲に、ＬＳＮ型読取り／書込みインタフェースを提供してよい。いくつかの実施形態では、各セグメントはセグメントＵＵＩＤ（例えば、セグメントの汎用一意識別子）によって識別されてよい。

【００４９】

記憶ページ：記憶ページは、概して固定サイズのメモリのブロックである。いくつかの実施形態では、各ページは、オペレーティングシステムによって定義されるサイズのメモリの（例えば、バーチャルメモリ、ディスク、または他の物理メモリの）ブロックであり、本明細書では用語「データブロック」によって参照されてもよい。すなわち、記憶ページは隣接セクタのセットであってよい。記憶ページは、ヘッダ及びメタデータがあるログページでの単位だけではなく、ＳＳＤでの割当ての単位としても役立ってよい。いくつかの実施形態では、及び本明細書に説明されるデータベースシステムの文脈では、用語「ページ」または「記憶ページ」は、通常、４０９６バイト、８１９２バイト、１６３８４バイト、または３２７６８バイト等の２の倍数であってよいデータベース構成によって定義されるサイズの類似したブロックを指してよい。

【００５０】

ログページ：ログページは、ログレコード（例えば、リドゥログレコードまたはアンドゥログレコード）を記憶するために使用される記憶ページのタイプである。いくつかの実施形態では、ログページは、サイズが記憶ページと同一であってよい。各ログページは、例えばそれが属するセグメントを識別するメタデータ等、そのログページについてのメタ

データを含むヘッダを含んでよい。ログページが編成の単位であり、必ずしも書込み動作に含まれるデータの単位ではないことがあることに留意されたい。例えば、いくつかの実施形態では、標準的な転送処理の間、書込み動作は、一度の１つのセクタをログの末尾に書き込んでよい。

【００５１】

ログレコード：ログレコード（例えば、ログページの個々の要素）はいくつかの異なるクラスであってよい。例えば、ストレージシステムのユーザー／クライアント／アプリケーションによって作成され、理解されるユーザーログレコード（ＵＬＲ）は、ボリューム内のユーザーデータに対する変更を示すために使用されてよい。ストレージシステムによって生成される制御ログレコード（ＣＬＲ）は、現在の無条件ボリューム耐久性（ｕｎｃｏｎｄｉｔｉｏｎａｌ　ｖｏｌｕｍｅ　ｄｕｒａｂｌｅ）ＬＳＮ（ＶＤＬ）等のメタデータを追跡調査するために使用される制御情報を含んでよい。ヌルログレコード（ＮＬＲ）は、いくつかの実施形態では、ログセクタまたはログページの未使用のスペースを充填するためのパディングとして使用されてよい。いくつかの実施形態では、これらのクラスのそれぞれの中に多様なタイプのログレコードがあってよく、ログレコードのタイプはログレコードを解釈するために呼び出される必要がある関数に対応してよい。例えば、１つのタイプは特定の圧縮フォーマットを使用する圧縮フォーマットのユーザーページのすべてのデータを表してよく、第２のタイプは、ユーザーページの中のバイト範囲の新しい値を表してよく、第３のタイプは、整数として解釈されるバイトのシーケンスに対する増分動作を表してよく、第４のタイプはページの中の別の場所に１バイト範囲をコピーすることを表してよい。いくつかの実施形態では、特にＵＬＲの場合、ログレコードタイプは、（整数または列挙型によってよりむしろ）バージョニング及び開発を簡略化してよいＧＵＩＤによって識別されてよい。

【００５２】

ペイロード：ログレコードのペイロードは、ログレコードに、または特定のタイプのログレコードに特有であるデータまたはパラメータ値である。例えば、いくつかの実施形態では、大部分（またはすべての）ログレコードが含み、ストレージシステム自体が理解するパラメータまたは属性のセットがあってよい。これらの属性は、セクタサイズに比較して相対的に小さくてよい共通のログレコードヘッダ／構造の部分であってよい。さらに、大部分のログレコードは、そのログレコードタイプに特有の追加のパラメータまたはデータを含んでよく、この追加情報はそのログレコードのペイロードと見なされてよい。いくつかの実施形態では、特定のＵＬＲのペイロードがユーザーページサイズよりも大きい場合、ペイロードは、そのペイロードがユーザーページのためのすべてのデータを含む絶対ＵＬＲ（ＡＵＬＲ）によって置き換えられてよい。これは、ストレージシステムがユーザーページのサイズに等しいＵＬＲのペイロードのサイズに対する上限を課すことができるようにしてよい。

【００５３】

セグメントログでログレコードを記憶する際に、いくつかの実施形態では、ペイロードはログヘッダとともに記憶されてよいことに留意されたい。他の実施形態では、ペイロードは別の場所に記憶されてよく、そのペイロードが記憶される場所に対するポインタはログヘッダとともに記憶されてよい。さらに他の実施形態では、ペイロードの一部はヘッダに記憶されてよく、ペイロードの残りは別個の場所に記憶されてよい。ペイロード全体がログヘッダとともに記憶される場合、これは帯域内ストレージと呼ばれてよい。それ以外の場合、ストレージは帯域外であると呼ばれてよい。いくつかの実施形態では、大部分の大きなＡＵＬＲのペイロードは（以下に説明される）ログのコールドゾーンで帯域外で記憶されてよい。

【００５４】

ユーザーページ：ユーザーページは、（固定サイズの）バイト範囲、及びストレージシステムのユーザー／クライアントに可視である特定のボリュームのためのそのアラインメントである。ユーザーページは論理概念であり、特定のユーザーページのバイトは任意の

記憶ページにそのまま記憶されてよい、または記憶されないことがある。特定のボリュームのユーザーページのサイズは、そのボリュームの記憶ページサイズとは無関係であってよい。いくつかの実施形態では、ユーザーページサイズはボリュームごとに設定可能であってよく、ストレージノード上の異なるセグメントは異なるユーザーページサイズを有してよい。いくつかの実施形態では、ユーザーページサイズは、セクタサイズ（例えば、４ＫＢ）の倍数となるように制約されてよく、上限（例えば、６４ＫＢ）を有してよい。他方、記憶ページサイズは、ストレージノード全体にとって固定であってよく、基礎的なハードウェアに対する変更がない限り変化しないことがある。

【００５５】

データページ：データページは、圧縮された形式でユーザーページデータを記憶するために使用される記憶ページのタイプである。いくつかの実施形態では、データページに記憶されるあらゆる１個のデータがログレコードと関連付けられ、各ログレコードは（データセクタとも呼ばれる）データページの中のセクタに対するポインタを含んでよい。いくつかの実施形態では、データページは各セクタによって提供されるメタデータ以外の任意の埋込みメタデータを含まないことがある。データページ内のセクタ間には関係性がなくてよい。代わりに、ページへの編成は、セグメントへのデータの割当ての粒度の表現としてのみ存在してよい。

【００５６】

ストレージノード：ストレージノードは、ストレージノードサーバコードが配備される単一のバーチャルマシンである。各ストレージノードは、複数のローカルにアタッチされたＳＳＤを含んでよく、１つまたは複数のセグメントへのアクセスにネットワークＡＰＩを提供してよい。いくつかの実施形態では、多様なノードはアクティブリスト上または（例えば、ノードが応答するには低速である、またはそれ以外の場合、正常に機能しないが、完全に使用不可ではない場合等）劣化したリスト上にあってよい。いくつかの実施形態では、クライアント側ドライバは、ノードが交換されるべきかどうか、及びいつノードが交換されるべきかを判断するため、及び／または観察された性能に基づいて、いつ及びどのようにして多様なノードの間でデータを再配分するのかを決定するために、ノードをアクティブまたは劣化として分類するのを支援してよい（または、分類するのに責任を負ってよい）

【００５７】

ＳＳＤ：本明細書において参照されるように、用語「ＳＳＤ」は、例えばディスク、ソリッドステートドライブ、電池によって支援されるＲＡＭ、不揮発性ＲＡＭデバイス（例えば、１つまたは複数のＮＶ－ＤＩＭＭ）、または別のタイプの永続ストレージデバイス等の、その記憶ボリュームによって利用されるストレージのタイプに関わりなく、ストレージノードによって見られるローカルブロック記憶ボリュームを指してよい。ＳＳＤは、必ずしも直接的にハードウェアにマッピングされない。例えば、異なる実施形態では、単一のソリッドステートストレージデバイスは、各ボリュームが複数のセグメントに分割され、複数のセグメントに渡ってストライピングされる複数のローカルボリュームに分けられる可能性がある、及び／または単一ドライブは単に管理の容易さのために複数のボリュームに分割されてよい。いくつかの実施形態では、各ＳＳＤは単一の固定場所で割当てマップを記憶してよい。このマップは、特定のセグメントによってどの記憶ページが所有されているのか、及び（データページと対照的に）これらのページの内のどれがログページであるのかを示してよい。いくつかの実施形態では、記憶ページは、転送処理が割当てを待機する必要がなくてよいように各セグメントに事前に割り当てられてよい。割当てマップに対するあらゆる変更は、新規に割り当てられた記憶ページがセグメントによって使用される前に耐久的にされる必要があることがある。

【００５８】

分散型データベース最適化ストレージシステムの一実施形態は、図４のブロック図によって示される。この例では、データベースシステム４００は、相互接続４６０上でデータベースエンジンヘッドノード４２０と通信する分散型データベース最適化ストレージシス

テム４１０を含む。図３に示される例でのように、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を含んでよい。この例では、分散型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として示されるストレージシステムサーバノードを含んだ）複数のストレージシステムサーバノードを含み、複数のストレージシステムサーバノードのそれぞれは、それが記憶するセグメント（複数の場合がある）のためのデータページ及びリドゥログのストレージ、多様なセグメント管理機能を実行するように構成されるハードウェア及び／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、つまり、複製（例えば、ストレージノードの中で等ローカルに）、データページを生成するためのリドゥログの合体、スナップショット（例えば、作成、復元、削除等）、ログ管理（例えば、ログレコードの操作）、クラッシュ回復、及び／または（例えば、セグメントの）スペース管理の内のいずれかまたはすべての少なくとも一部を実行するように構成されるハードウェア及び／またはソフトウェアを含んでよい。各ストレージシステムサーバノードは、データブロックがクライアント（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加入者）の代わりに記憶されてよい（例えば、ＳＳＤ等の）複数のアタッチされたストレージデバイスも有してよい。

【００５９】

図４に示される例では、ストレージシステムサーバノード４３０は、データページ（複数の場合がある）４３３、セグメントリドゥログ（複数の場合がある）４３５、セグメント管理機能４３７、及びアタッチされたＳＳＤ４７１から４７８を含む。再び、ラベル「ＳＳＤ」はソリッドステートドライブを指してよい、または指さないこともあるが、基礎的なハードウェアに関わりなく、より概してローカルブロック記憶ボリュームを指してよいことに留意されたい。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（複数の場合がある）４４３、セグメントリドゥログ（複数の場合がある）４４５、セグメント管理機能４４７、及びアタッチされたＳＳＤ４８１から４８８を含み、ストレージシステムサーバノード４５０は、データページ（複数の場合がある）４５３、セグメントリドゥログ（複数の場合がある）４５５、セグメント管理機能４５７、及びアタッチされたＳＳＤ４９１から４９８を含む。

【００６０】

上述されたように、いくつかの実施形態では、セクタは、ＳＳＤでのアラインメントの単位であり、書込みが部分的だけに完了されるリスクなしに書き込むことができるＳＳＤでの最大サイズであってよい。例えば、多様なソリッドステートドライブ及びスピニングメディアのセクタサイズは４ＫＢであってよい。本明細書に説明される分散型データベース最適化ストレージシステムのいくつかの実施形態では、ありとあらゆるセクタは、セクタがその一部であるより高レベルのエンティティに関わりなく、セクタの始まりに６４ビット（８バイト）のＣＲＣを含んで有してよい。係る実施形態では、（セクタがＳＳＤから読み取られるたびに確証されてよい）このＣＲＣは破損を検出する際に使用されてよい。いくつかの実施形態では、ありとあらゆるセクタは、その値がセクタをログセクタ、データセクタ、または初期化されていないセクタとして該セクタを識別する「セクタタイプ」バイトを含んでもよい。例えば、いくつかの実施形態では、０のセクタタイプバイト値は、セクタが初期化されていないことを示してよい。

【００６１】

いくつかの実施形態では、分散型データベース最適化ストレージシステムのストレージシステムサーバノードのそれぞれは、例えばリドゥログを受信し、データページ等を送り返すために、データベースエンジンヘッドノードとの通信を管理するノードサーバのオペレーティングシステムで実行中のプロセスのセットを実装してよい。いくつかの実施形態では、分散型データベース最適化ストレージシステムに書き込まれるすべてのデータブロックは、（例えば、リモートキー値耐久性バックアップストレージシステムで）長期の及び／またはアーカイブのストレージにバックアップされてよい。

【００６２】

図５は、一実施形態に係る、データベースシステムでの別個の分散型データベース最適化ストレージシステムの使用を示すブロック図である。この例では、１つまたは複数のクライアントプロセス５１０が、データベースエンジン５２０及び分散型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維持される１つまたは複数のデータベースにデータを記憶してよい。図５に示される例では、データベースエンジン５２０がデータベース階層構成要素５６０、及び（分散型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構成要素５６０との間のインタフェースとして働く）クライアント側ドライバ５４０を含む。いくつかの実施形態では、データベース階層構成要素５６０は、図３のクエリーパーシング、最適化、及び実行構成要素３０５、並びにトランザクション及び一貫性管理構成要素３３０によって実行される機能等の機能を実行してよい、及び／またはデータページ、トランザクションログ、及び／またはアンドゥログ（例えば、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、及びアンドゥログ３４５によって記憶されるもの）を記憶してよい。

【００６３】

この例では、１つまたは複数のクライアントプロセス５１０は、データベース階層構成要素５６０に（ストレージノード５３５ａから５３５ｎの内の１つまたは複数に記憶されるデータをターゲットとする読取り要求及び／または書込み要求を含んでよい）データベースクエリー要求５１５を送信してよく、データベース階層構成要素５６０からデータベースクエリー応答５１７（例えば、書込み肯定応答及び／または要求されたデータを含む応答）を受信してよい。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ー要求５１５は、分散型データベース最適化ストレージシステム５３０への以後のルーティングのためにクライアント側ドライバ５４０に送信されてよい、１つまたは複数のレコード書込み要求５４１を生成するためにパースされ、最適化されてよい。この例では、クライアント側ドライバ５４０は、それぞれのレコード書込み要求５４１に対応する１つまたは複数のリドゥログレコード５３１を生成してよく、リドゥログレコード５３１を分散型データベース最適化ストレージシステム５３０のストレージノード５３５の特定のストレージノードに送信してよい。分散型データベース最適化ストレージシステム５３０は、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）各リドゥログレコード５３１の対応する書込み肯定応答５３２を返してよい。クライアント側ドライバ５４０は、これらの書込み肯定応答をデータベース階層構成要素５６０に（書込み応答５４２として）渡してよく、データベース階層構成要素５６０は次いでデータベースクエリー応答５１７の内の１つとして１つまたは複数のクライアントプロセス５１０に対応する応答（例えば、書込み肯定応答）を送信してよい。

【００６４】

この例では、データページを読み込む要求を含む各データベースクエリー要求５１５は、１つまたは複数のレコード読取り要求５４３を生成するためにパースされ、最適化されてよく、レコード読取り要求５４３は分散型データベース最適化ストレージシステム５３０への以後のルーティングのためにクライアント側ドライバ５４０に送信されてよい。この例では、クライアント側ドライバ５４０は、分散型データベース最適化ストレージシステム５３０のストレージノード５３５の特定のストレージノードにこれらの要求を送信してよく、分散型データベース最適化ストレージシステム５３０は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）要求されたデータページ５３３を返してよい。クライアント側ドライバ５４０は、戻りデータレコード５４４としてデータベース階層構成要素５６０に返されたデータページを送信してよく、データベース階層構成要素５６０は次いでデータベースクエリー応答５１７として１つまたは複数のクライアントプロセス５１０にデータページを送信してよい。

【００６５】

いくつかの実施形態では、多様なエラーメッセージ及び／またはデータ損失メッセージ５３４が、分散型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）送信されてよい。これらの

メッセージは、クライアント側ドライバ５４０から、エラー報告メッセージ及び／または損失報告メッセージ５４５として、データベース階層構成要素５６０に、及び次いで１つまたは複数のクライアントプロセス５１０に、データベースクエリー応答５１７とともに（または代わりに）渡されてよい。

【００６６】

いくつかの実施形態では、分散型データベース最適化ストレージシステム５３０のＡＰＩ５３１から５３４、及びクライアント側ドライバ５４０のＡＰＩ５４１から５４５は、データベースエンジン５２０が分散型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かのように、分散型データベース最適化ストレージシステム５３０の機能性をデータベースエンジン５２０に曝露してよい。例えば、データベースエンジン５２０は、データベースエンジン５２０及び分散型データベース最適化ストレージシステム５３０の組合せによって実装されるデータベースシステムの多様な動作（例えば、記憶動作、アクセス動作、ロギング変更動作、回復動作、及び／またはスペース管理動作）を実行するために（またはそれらの実行を容易にするために）（クライアント側ドライバ５４０を通して）リドゥログレコードまたは要求データページをこれらのＡＰＩを通して書き込んでよい。図５に示されるように、分散型データベース最適化ストレージシステム５３０は、それぞれが複数のアタッチされたＳＳＤを有してよいストレージノード５３５ａから５３５ｎにデータブロックを記憶してよい。いくつかの実施形態では、分散型データベース最適化ストレージシステム５３０は、多様なタイプの冗長性方式の適用によって、記憶されているデータブロックに高い耐久性を提供してよい。

【００６７】

多様な実施形態では、図５のデータベースエンジン５２０と分散型データベース最適化ストレージシステム５３０との間のＡＰＩ呼出し及び応答（例えば、ＡＰＩ５３１から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構成要素５６０との間のＡＰＩ呼出し及び応答（例えば、ＡＰＩ５４１から５４５）は、（例えば、ゲートウェイ制御プレーンによって管理される）安全なプロキシ接続上で実行されてよい、または公衆ネットワーク上でもしくは代わりにバーチャルプライベートネットワーク（ＶＰＮ）接続等のプライベートチャネル上で実行されてよいことに留意されたい。本明細書に説明されるデータベースシステムの構成要素への、及びデータベースシステムの構成要素の間のこれらの及び他のＡＰＩは、シンプルオブジェクトアクセスプロトコル（ＳＯＡＰ）技術及び表象状態転送（ＲＥＳＴ）技術を含むが、これに限定されるものではない異なる技術に従って実装されてよい。例えば、これらのＡＰＩは、ＳＯＡＰ ＡＰＩまたはＲＥＳＴｆｕｌ ＡＰＩとして実装されてよいが、必ずしも実装されない。ＳＯＡＰは、ウェブベースのサービスとの関連で情報を交換するためのプロトコルである。ＲＥＳＴは分散型ハイパーメディアシステム用のアーキテクチャスタイルである。（ＲＥＳＴｆｕｌウェブサービスとも呼ばれてよい）ＲＥＳＴｆｕｌ ＡＰＩは、ＨＴＴＰ及びＲＥＳＴ技術を使用して実装されるウェブサービスＡＰＩである。本明細書に説明されるＡＰＩは、いくつかの実施形態では、データベースエンジン５２０及び／または分散型データベース最適化ストレージシステム５３０との統合をサポートするために、Ｃ、Ｃ＋＋、Ｊａｖａ、Ｃ＃、及びＰｅｒｌを含むが、これに限定されるものではない多様な言語でクライアントライブラリでラップされてよい。

【００６８】

上述されたように、いくつかの実施形態では、データベースシステムの機能構成要素は、データベースエンジンによって実行される構成要素と、別個の分散されたデータベース最適化ストレージシステムで実行される構成要素との間で仕切られてよい。１つの特定の例では、（例えば、単一のデータブロックを、そのデータブロックにレコードを追加することによって更新するために）何かをデータベースに挿入する要求をクライアントプロセス（またはクライアントプロセスのスレッド）から受信することに応えて、データベースエンジンヘッドノードの１つまたは複数の構成要素は、クエリーパーシング、最適化、及び実行を実行してよく、クエリーの各部分をトランザクション及び一貫性管理構成要素に

送信してよい。トランザクション及び一貫性管理構成要素は、他のクライアントプロセス（またはクライアントプロセスのスレッド）が同時に同じ行を修正しようとしていないことを保証してよい。例えば、トランザクション及び一貫性管理構成要素は、この変更がデータベースにおいて原子的に、一貫して、耐久的に、及び独立して実行されることを保証することに責任を負ってよい。例えば、トランザクション及び一貫性管理構成要素は、分散型データベース最適化ストレージサービスのノードの１つに送信されるリドゥログレコードを生成し、ＡＣＩＤプロパティがこのトランザクションについて満たされていることを保証する順序で及び／またはタイミングでリドゥログレコードを（他のクライアント要求に応えて生成される他のリドゥログとともに）分散型データベース最適化ストレージサービスに送信するために、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ドライバとともに機能してよい。対応するストレージノードは、（ストレージサービスによって「更新レコード」と見なされてよい）リドゥログレコードを受信すると、データブロックを更新し、データブロックのリドゥログを更新してよい（例えば、データブロックに向けられるすべての変更のレコード）。いくつかの実施形態では、データベースエンジンは、この変更のためにアンドゥログレコードを生成することに責任を負ってよく、アンドゥログのためのリドゥログレコードを生成することにも責任を負ってよく、この両方ともトランザクション性を保証するために（データベース階層で）ローカルに使用されてよい。ただし、従来のデータベースシステムにおいてとは異なり、本明細書に説明されるシステムは、（変更をデータベース階層で適用し、修正されたデータブロックをストレージシステムに送るよりむしろ）データブロックに変更を適用するための責任をストレージシステムに移してよい。さらに、図８から図９Ｂで本明細書に説明されるように、多様な実施形態では、システム全体のチェックポイントは、ストレージシステムによっても実行されてよい多様なログレコード演算に起因するデータベースシステムクラッシュからの高速回復とともに、データベースシステムで回避されてよい。

【００６９】

異なる実施形態で、さまざまな割当てモデルがＳＳＤのために実装されてよい。例えば、いくつかの実施形態では、ログエントリページ及び物理アプリケーションページが、ＳＳＤデバイスと関連付けられたページの単一のヒープから割り当てられてよい。この手法は、未指定のままとなるために、及び自動的に使用に適合するためにログページ及びデータページによって消費される相対的な記憶量を残すという優位点を有してよい。また、手法は、ページが使用され、準備なしに随意に転用されるまでページを準備されないままにできるという優位点も有してよい。他の実施形態では、割当てモデルはストレージデバイスをログエントリ及びデータページのための別々のスペースに仕切ってよい。一度係る割当てモデルが図６のブロック図に示され、以下に説明される。

【００７０】

図６は、一実施形態に係る、分散型データベース最適化ストレージシステムの所与のストレージノード（または永続ストレージデバイス）にデータ及びメタデータがどのように記憶されてよいのかを示すブロック図である。この例では、ＳＳＤストレージスペース６００は、６１０と名前が付けられたスペースの部分にＳＳＤヘッダ及び他の固定メタデータを記憶する。ＳＳＤストレージスペース６００は、６２０と名前が付けられたスペースの部分にログページを記憶し、追加のログページのために初期化され、確保される、６３０と名前が付けられたスペースを含む。（６４０として示される）ＳＳＤストレージスペース６００の一部分は初期化されているが、割り当てられておらず、（６５０として示される）スペースの別の部分は初期化されておらず、割り当てられていない。最後に、６６０と名前が付けられたＳＳＤストレージスペース６００の部分はデータページを記憶する。

【００７１】

この例では、最初の使用可能なログページスロットは６１５として示され、最後の使用されたログページスロット（一時的）は６２５として示される。最後の確保されたログページスロットは６３５として示され、最後の使用可能なログページスロットは６４５とし

て示される。この例では、最初の使用されたデータページスロット（一時的）は６６５として示される。いくつかの実施形態では、ＳＳＤストレージスペース６００の中でのこれらの要素（６１５、６２５、６３５、６４５、及び６６５）のそれぞれの位置は、それぞれのポインタによって識別されてよい。

【００７２】

図６に示される割当て手法では、有効なログページはフラットストレージスペースの始まりにパックされてよい。ログページが解放されるために開く穴は、アドレススペースのさらに先に入る追加のログページスロットが使用される前に再使用されてよい。例えば、最悪の場合、最初のｎ個のログページスロットが有効なログデータを含み、この場合、ｎは今まで同時に存在した有効なログページの最大数である。この例では、有効データページはフラットストレージスペースの最後にパックされてよい。データページが解放されることにより開く穴は、アドレススペースでより下方の追加のデータページスロットが使用される前に再使用されてよい。例えば、最悪の場合、最後のｍのデータページが有効なデータを含み、この場合ｍは今まで同時に存在した有効なデータページの最大数である。

【００７３】

いくつかの実施形態では、ログページスロットが有効なログページエントリの潜在的なセットの部分になることができる前に、ログページスロットは有効な将来のログエントリページのために混同できない値に初期化される必要がある。廃棄されたログページは新しい有効なログページについて絶対に混同されることがないほど十分なメタデータを有するので、これは、リサイクルされるログページスロットに暗黙に当てはまる。ただし、ストレージデバイスが最初に初期化されるとき、またはアプリケーションデータページを記憶するために潜在的に使用されたスペースが再利用されるとき、ログページスロットは、ログページスロットがログページスロットプールに加えられる前に初期化される必要がある。いくつかの実施形態では、ログスペースのバランスを取り戻す／再利用することは、バックグラウンドタスクとして実行されてよい。

【００７４】

図６に示される例では、カレントログページスロットプールは（６１５で）最初の使用可能なログページスロットと最後の確保されたログページスロット（６２５）との間に領域を含む。いくつかの実施形態では、このプールは、（例えば、最後の確保されたログページスロット６３５を識別するポインタに対する更新を持続させることによって）新しいログページスロットの再初期化なしに最後の使用可能なログページスロット（６２５）まで安全に増大してよい。この例では、（ポインタ６４５によって識別される）最後の使用可能なログページスロットを超えて、プールは、初期化されたログページスロットを持続し、最後の使用可能なログページスロット（６４５）のためのポインタを持続的に更新することによって、（ポインタ６６５によって識別される）最初の使用されたデータページスロットまで成長してよい。この例では、６５０として示される、ＳＳＤストレージスペース６００の以前に初期化されておらず、割り当てられていない部分は、ログページを記憶するためにとりあえず利用されてよい。いくつかの実施形態では、カレントログページスロットプールは、最後の確保されたログページスロット（６３５）のポインタに対する更新を持続することによって（ポインタによって識別される）最後の使用されたログページスロットの位置まで縮小されてよい。

【００７５】

図６に示される例では、カレントデータページスロットプールは、（ポインタ６４５によって識別される）最後の使用可能なログページスロットと、ＳＳＤストレージスペース６００の最後との間に領域を含む。いくつかの実施形態では、データページプールは、最後の使用可能なログページスロット（６４５）のポインタに対する更新を持続するによって、最後の確保されたログページスロット（６３５）に対するポインタによって識別される位置まで安全に成長してよい。この例では、６４０として示される、ＳＳＤストレージスペース６００の以前に初期化されたが、割り当てられていない部分は、データページを記憶するためにとりあえず利用されてよい。これを超えて、プールは、最後の確保された

ログページスロット（６３５）及び最後の使用可能なログページスロット（６４５）のポインタに対する更新を持続し、ログページよりむしろデータページを記憶するために、６３０及び６４０として示されるＳＳＤストレージスペース６００の部分を効果的に割り当てし直すことによって、最後の使用されたログページスロット（６２５）のポインタによって識別される位置まで安全に成長してよい。いくつかの実施形態では、データページスロットプールは、追加のログページスロットを初期化し、最後の使用可能なログページスロット（６４５）のポインタに対する更新を持続することによって、最初の使用されたデータページスロット（６６５）のポインタによって識別される位置まで安全に縮小されてよい。

【００７６】

図６に示される割当て手法を利用する実施形態では、ログページプール及びデータページプールのページサイズは、優れたパッキング挙動を容易にしつつも、独立して選択されてよい。係る実施形態では、有効なログページが、アプリケーションデータによって形成されるスプーフィングされたログページにリンクする可能性はないことがあり、壊れたログと依然として書き込まれていない次のページにリンクする有効なログテールとを区別することが可能なことがある。図６に示される割当て手法を利用する実施形態では、起動時、最後の確保されたログページスロット（６３５）に対するポインタによって識別される位置までのログページスロットのすべてが迅速に且つ連続して読み取られてよく、（推論されるリンキング／順序付けを含む）ログインデックス全体が再構築されてよい。係る実施形態では、すべてはＬＳＮ順序制御制約から推論できるので、ログページ間の明示的なリンキングの必要性がないことがある。

【００７７】

いくつかの実施形態では、セグメントは３つの主要な部分（またはゾーン）、つまり、ホットログを含む部分、コールドログを含む部分、及びユーザーページデータを含む部分から構成されてよい。ゾーンは、必ずしもＳＳＤの隣接領域ではない。むしろ、ゾーンは、記憶ページの粒度で点在することがある。さらに、セグメント及びそのプロパティについてのメタデータを記憶するセグメントごとにルートページがあってよい。例えば、セグメントのルートページはセグメントのためのユーザーページサイズ、セグメント内のユーザーページの数、（フラッシュ番号（ｆｌｕｓｈ　ｎｕｍｂｅｒ）の形で記録されてよい）ホットログゾーンの現在の始まり／ヘッド、ボリュームエポック、及び／またはアクセス制御メタデータを記憶してよい。

【００７８】

いくつかの実施形態では、ホットログゾーンは、それらがストレージノードによって受信されるにつれ、クライアントからの新しい書込みを受け入れてよい。ページの以前のバージョンからのデルタの形をとるユーザーページ／データページに対する変更を指定するデルタユーザーログレコード（ＤＵＬＲ）及び完全なユーザーページ／データページのコンテンツを指定する絶対ユーザーログレコード（ＡＵＬＲ）の両方とも、ログに完全に書き込まれてよい。ログレコードは、ほぼ、ログレコードが受信される順序でこのゾーンに追加されてよく（例えば、ログレコードがＬＳＮでソートされるのではない）、それらはログページに渡って広がることがある。例えばログレコードは独自のサイズの表示を含んでよい等、ログレコードは自己記述的である必要がある。いくつかの実施形態では、ガベージコレクションはこのゾーンで実行されない。代わりに、スペースは、すべての必要とされるログレコードがコールドログにコピーされた後にログの始まりから切り詰めることによって再利用されてよい。ホットゾーンのログセクタは、セクタが作成されるたびに最も最近の既知の無条件ＶＤＬで注釈されてよい。条件付きのＶＤＬ　ＣＬＲは、それらが受信されるにつれホットゾーンに書き込まれてよいが、最も最近に書き込まれたＶＤＬ　ＣＬＲだけが意味を持ってよい。

【００７９】

いくつかの実施形態では、新しいログページが書き込まれるたびに、新しいログページにはフラッシュ番号が割り当てられる。フラッシュ番号は、各ログページの中のあらゆる

セクタの部分として書き込まれてよい。フラッシュ番号は、２つのログページを比較するときに、どのログページが後に書き込まれたのかを決定するために使用されてよい。フラッシュ番号は単調に増加し、ＳＳＤ（またはストレージノード）に対して調べられて（ｓｃｏｐｅｄ）よい。例えば、単調に増加するフラッシュ番号のセットは、ＳＳＤ上のすべてのセグメント（またはストレージノード上のすべてのセグメント）の間で共有される。

【００８０】

いくつかの実施形態では、コールドログゾーンで、ログレコードはそのＬＳＮの昇順で記憶されてよい。このゾーンでは、ＡＵＬＲはそのサイズに応じて必ずしもデータをインラインで記憶しないことがある。例えば、ＡＵＬＲが大きなペイロードを有する場合、ペイロードのすべてまたは一部がデータゾーンに記憶されてよく、ＡＵＬＲはそのデータがデータゾーンのどこに記憶されているのかを指してよい。いくつかの実施形態では、コールドログゾーンのログページは、セクタ単位でよりむしろ、一度に１全ページ、書き込まれてよい。コールドゾーンのログページは一度に全ページ書き込まれるため、全セクタ内のフラッシュ番号が同一ではないコールドゾーンのどのようなログページも不完全に書き込まれたページと見なされてよく、無視されてよい。いくつかの実施形態では、コールドログゾーンでは、ＤＵＬＲは（最大２ログページまで）複数のログページに及ぶことができることがある。しかし、ＡＵＬＲは、例えば合体動作が単一の原子的な書込みでＤＵＬＲをＡＵＬＲで置き換えることができるように、複数のログセクタに及ぶことができないことがある。

【００８１】

いくつかの実施形態では、コールドログゾーンは、ホットログゾーンからログレコードをコピーすることによってポピュレートされる。係る実施形態では、ＬＳＮが現在の無条件ボリューム耐久性ＬＳＮ（ＶＤＬ）以下であるログレコードだけがコールドログゾーンにコピーされる資格があってよい。ホットログゾーンからコールドログゾーンにログレコードを移動するとき、（多くのＣＬＲ等の）いくつかのログレコードは、それらがもはや必要ではないため、コピーされる必要がないことがある。さらにユーザーページのなんらかの追加の合体がこの点で実行されてよく、このことが必要とされるコピーの量を削減してよい。いくつかの実施形態では、いったん所与のホットゾーンログページが完全に書き込まれ、もはや最新のホットゾーンログページではなく、ホットゾーンログページ上のすべてのＵＬＲがコールドログゾーンに無事にコピーされると、ホットゾーンログページは解放され、再使用されてよい。

【００８２】

いくつかの実施形態では、例えば記憶階層のＳＳＤにもはや記憶される必要のないログレコード等、もはやサポートされていないログレコードによって占められているスペースを再利用するために、ガベージコレクションがコールドログゾーンで行われてよい。例えば、ログレコードは同じユーザーページに対する以後のＡＵＬＲがあるときにサポートされなくなってよく、ログレコードによって表されるユーザーページのバージョンはＳＳＤでの保持に必要とされない。いくつかの実施形態では、ガベージコレクションプロセスは、２つ以上の隣接するログページをマージし、２つ以上の隣接するログページをそれらのページが置き換えているログページからの旧式ではないログレコードのすべてを含むより少ない新しいログページで置き換えることによってスペースを再利用してよい。新しいログページには、それらが置き換えているログページのフラッシュ番号よりも大きい新しいフラッシュ番号が割り当てられてよい。これらの新しいログページの書込みが完了した後に、置き換えられたログページが空きページプールに加えられてよい。いくつかの実施形態では、あらゆるポインタを使用するログページの明示的な連鎖がないことがあることに留意されたい。代わりに、ログページのシーケンスはそれらのページに対するフラッシュ番号によって暗黙に決定されてよい。ログレコードの複数のコピーが検出されるたびに、最高のフラッシュ番号のログページに存在するログレコードが有効であると見なされてよく、他はもはやサポートされないと見なされてよい。

【００８３】

いくつかの実施形態では、例えば、データゾーン（セクタ）の中で管理されるスペースの粒度がデータゾーン（記憶ページ）の外の粒度とは異なってよいため、なんらかのフラグメンテーションがあってよい。いくつかの実施形態では、このフラグメンテーションを管理するために、システムは各データページによって使用されるセクタの数を追跡調査してよく、ほぼ全データページから優先的に割り当ててよく、（データを新しい場所に、それが依然として関連している場合に移動することを必要としてよい）ほぼ空のデータページのガベージコレクションを優先的に行ってよい。セグメントに割り当てられるページが、いくつかの実施形態では３つのゾーンの間で転用されてよいことに留意されたい。例えば、セグメントに割り当てられていたページが解放されると、ページはある期間そのセグメントと関連付けられたままとなってよく、後にそのセグメントの３つのゾーンのいずれかで使用されてよい。あらゆるセクタのセクタヘッダは、セクタが属するゾーンを示してよい。いったんページ内のすべてのセクタが空くと、ページは、ゾーンに渡って共有される共通の空き記憶ページプールに返されてよい。この空き記憶ページの共有は、いくつかの実施形態では、フラグメンテーションを削減（または回避）してよい。

【００８４】

いくつかの実施形態では、本明細書に説明される分散型データベース最適化ストレージシステムは、メモリ内に多様なデータ構造を維持してよい。例えば、セグメントに存在するユーザーページごとに、ユーザーページテーブルが、このユーザーページが「クリアされる」かどうか（つまり、このユーザーページがすべてのゼロを含んでいるかどうか）、該ページのためのコールドログゾーンからの最新のログレコードのＬＳＮ、及びページのホットログゾーンからのすべてのログレコードの場所のアレイ／リストを示すビットを記憶してよい。ログレコードごとに、ユーザーページテーブルはセクタ番号、そのセクタの中のログレコードのオフセット、そのログページの中で読み取るセクタの数、（ログレコードが複数のログページに及ぶ場合）第２のログページのセクタ番号、及びそのログページの中で読み取るセクタの数を記憶してよい。いくつかの実施形態では、ユーザーページテーブルは、コールドログゾーンからのあらゆるログレコードのＬＳＮ、及び／またはＡＵＬＲがコールドログゾーンにある場合、最新のＡＵＬＲのペイロードのセクタ番号のアレイを記憶してもよい。

【００８５】

本明細書に説明される分散型データベース最適化ストレージシステムのいくつかの実施形態では、ＬＳＮインデックスはメモリに記憶されてよい。ＬＳＮインデックスは、コールドログゾーンの中のログページにＬＳＮをマッピングしてよい。コールドログゾーンのログレコードがソートされていることを考えれば、それはログページあたり１つのエントリを含むためであってよい。ただし、いくつかの実施形態では、あらゆる旧式ではないＬＳＮがインデックスに記憶され、対応するセクタ番号、オフセット、及びログレコードごとのセクタの数にマッピングされてよい。

【００８６】

本明細書に説明される分散型データベース最適化ストレージシステムのいくつかの実施形態では、ログページテーブルはメモリに記憶されてよく、ログページテーブルはコールドログゾーンのガベージコレクションの間に使用されてよい。例えば、ログページテーブルはどのログレコードがもはやサポートされていないのか（例えば、どのログレコードのガベージコレクションを行うことができるのか）、及び各ログページでどれほど多くの空きスペースが使用できるのかを識別してよい。

【００８７】

本明細書に説明されるストレージシステムでは、エクステントは、ボリュームを表すために他のエクステントと結合できる（連結できる、またはストライピングできるのかのどちらか）ストレージの高度に耐久性の単位を表す論理概念であってよい。各エクステントは、単一の保護グループでのメンバーシップによって耐久的にされてよい。エクステントは、ＬＳＮ型の読取り／書込みインタフェースを、作成時に定義される固定サイズを有する隣接バイトサブレンジに提供してよい。エクステントに対する読取り／書込み動作は、

含む側の保護グループによって１つまたは複数の適切なセグメント読取り／書込み動作にマッピングされてよい。本明細書に使用されるように、用語「ボリュームエクステント」は、ボリュームの中のバイトの特有のサブレンジを表すために使用されるエクステントを指してよい。

【００８８】

上述されたように、ボリュームは、それぞれが１つまたは複数のセグメントから構成される保護グループによって表される複数のエクステントから構成されてよい。いくつかの実施形態では、異なるエクステントに向けられるログレコードはインタリーブされたＬＳＮを有してよい。ボリュームに対する変更が特定のＬＳＮまで耐久的となるためには、そのＬＳＮまでのすべてのログレコードが、それらが属しているエクステントに関わりなく耐久的である必要があってよい。いくつかの実施形態では、クライアントは、まだ耐久的にされていない未決ログレコードを追跡調査してよく、いったん特定のＬＳＮまでのすべてのＵＬＲが耐久的にされると、クライアントはボリュームの保護グループの内の１つにボリューム耐久性ＬＳＮ（ＶＤＬ）メッセージを送信してよい。ＶＤＬは、保護グループのすべての同期ミラーセグメントに書き込まれてよい。これは「無条件ＶＤＬ」と呼ばれることがあり、それはセグメントで起こる書込み活動とともに多様なセグメントに（またはより詳細には、多様な保護グループに）周期的に持続されてよい。いくつかの実施形態では、無条件ＶＤＬはログセクタヘッダに記憶されてよい。

【００８９】

多様な実施形態では、セグメントで実行されてよい動作は、（ホットログゾーンの末尾にＤＵＬＲまたはＡＵＬＲを書き込み、次いでユーザーページテーブルを更新することを含んでよい）クライアントから受信されたＤＵＬＲまたはＡＵＬＲを書き込むこと、（ユーザーページのデータセクタの位置を突き止め、あらゆる追加のＤＵＬＲを適用する必要なしにデータセクタを返すことを含んでよい）コールドユーザーページを読み取ること、（ユーザーページの最も最新のＡＵＬＲのデータセクタの位置を突き止めることを含み、ユーザーページに、それを返す前にあらゆる以後のＤＵＬＲを適用してよい）ホットユーザーページを読み取ること、（適用された最後のＤＵＬＲを置き換えるＡＵＬＲを作成するためにユーザーページのＤＵＬＲを合体させることを含んでよい）ＤＵＬＲをＡＵＬＲで置き換えること、ログレコードを操作すること等を含んでよい。本明細書に説明されるように、合体は、ユーザーページのより最近のバージョンを作成するためにユーザーページの初期のバージョンにＤＵＬＲを適用するプロセスである。（別のＤＵＬＲが書き込まれるまで）合体の前に書き込まれたすべてのＤＵＬＲは要求に応じて読み取られ、適用される必要はないことがあるため、ユーザーページを合体させることは読取りレーテンシを削減するのに役立ってよい。また、合体は、（ログレコードが存在することを必要とするスナップショットがないならば）旧いＡＵＬＲ及びＤＵＬＲをもはやサポートされなくすることによってストレージスペースを再利用するのに役立ってよい。いくつかの実施形態では、合体動作は、最も最新のＡＵＬＲを場所を見つけ、ＤＵＬＲのいずれも省略することなく、あらゆる以後のＤＵＬＲを順番に適用することを含んでよい。上述されたように、いくつかの実施形態では、合体はホットログゾーンの中で実行されないことがある。代わりに、合体はコールドログゾーンの中で実行されてよい。いくつかの実施形態では、合体は、ログレコードがホットログゾーンからコールドログゾーンにコピーされるにつれて実行されてもよい。

【００９０】

いくつかの実施形態では、ユーザーページを合体させる決定は、（例えば、ＤＵＬＲチェーンの長さが合体動作の所定の閾値を超える場合、システム全体での方針、アプリケーション特有の方針、またはクライアントによって指定される方針に従って））、またはクライアントに読み取られているユーザーページごとに、ページの未決のＤＵＬＲチェーンのサイズによってトリガされてよい。

【００９１】

図７は、一実施形態に係る、データベースボリューム７１０の例の構成を示すブロック

図である。この例では、（アドレス範囲７１５ａから７１５ｅとして示される）多様なアドレス範囲７１５のそれぞれに対応するデータが（セグメント７４５ａから７４５ｎとして示される）異なるセグメント７４５として記憶される。すなわち、多様なアドレス範囲７１５のそれぞれに対応するデータは（エクステント７２５ａから７２５ｂ、及びエクステント７３５ａから７３５ｈとして示される）異なるエクステントに編成されてよく、これらのエクステントの多様なエクステントが、（ストライプセット７２０ａ及びストライプセット７２０ｂとして示されるもの等の）ストライピングを行って、または行わないで（７３０ａから７３０ｆとして示される）異なる保護グループ７３０に含まれてよい。この例では、保護グループ１はイレイジャーコーディングの使用を示す。この例では、保護グループ２及び３、並びに保護グループ６及び７は互いのミラーリングされたデータセットを表す。一方、保護グループ４は単一インスタンス（非冗長）データセットを表す。この例では、保護グループ８は、他の保護グループを結合する複数階層保護グループを表す（例えば、これは複数領域保護グループを表してよい）。この例では、ストライプセット１（７２０ａ）及びストライプセット２（７２０ｂ）は、いくつかの実施形態で、エクステント（例えば、エクステント７２５ａ及び７２５ｂ）がどのようにしてボリュームの中にストライピングされてよいのかを示す。

【００９２】

すなわち、この例では、保護グループ１（７３０ａ）は、それぞれ範囲１から３（７１５ａから７１５ｃ）のデータを含むエクステントａからｃ（７３５ａから７３５ｃ）を含み、これらのエクステントはセグメント１から４（７４５ａから７４５ｄ）にマッピングされる。保護グループ２（７３０ｂ）は、範囲４（７１５ｄ）からストライピングされたデータを含むエクステントｄ（７３５ｄ）を含み、このエクステントはセグメント５から７（７４５ｅから７４５ｇ）にマッピングされる。同様に、保護グループ３（７３０ｃ）は、範囲４（７１５ｄ）からストライピングされたデータを含むエクステントｅ（７３５ｅ）を含み、セグメント８から９（７４５ｈから７４５ｉ）にマッピングされ、保護グループ４（７３０ｄ）は、範囲４（７１５ｄ）からストライピングされたデータを含むエクステントｆ（７３５ｆ）を含み、セグメント１０（７４５ｊ）にマッピングされる。この例では、保護グループ６（７３０ｅ）は、範囲５（７１５ｅ）からストライピングされたデータを含むエクステントｇ（７３５ｇ）を含み、セグメント１１から１２（７４５ｋから７４５ｌ）にマッピングされ、保護グループ７（７３０ｆ）は、やはり範囲５（７１５ｅ）からストライピングされたデータを含むエクステントｈ（７３５ｈ）を含み、セグメント１３－１４（７４５ｍから７４５ｎ）にマッピングされる。

【００９３】

ここで図８を参照すると、多様な実施形態では、上述されたように、データベースシステムは、ストレージノードのデータページの中に記憶されているデータに対する多様なアクセス要求（例えば、書込み要求）に応えてリドゥログレコードを生成し、リドゥログレコードが生成されたそれぞれのデータページを記憶するストレージノードにリドゥログレコードを送信するように構成されてよい。ストレージノードは、特定のデータページのための合体イベントを検出し、それに応じて特定のデータページのために合体動作を実行してよい。典型的なデータベースシステムは、一方、周期的な間隔で記憶されるデータに適用される生成されたリドゥログのすべてをフラッシュし、このようにしてデータベースによって実行されるアクセス要求及び他のタスクの処理を中断させるシステム全体のチェックポイントを適用してよい。

【００９４】

図８の方法は、分散型データベース最適化ストレージシステム４１０（例えば、ストレージシステムサーバノード（複数の場合がある）４３０、４４０、４５０等）のログ構造化ストレージシステムの多様な構成要素によって実行されているとして説明されてよいが、方法はいくつかの場合、いずれの特定の構成要素によっても実行される必要はない。例えば、いくつかの場合、図８の方法は、いくつかの実施形態に従ってなんらかの他の構成要素またはコンピュータシステムによって実行されてよい。また、いくつかの場合、デー

タベースシステム４００の構成要素は、図４の例に示されるのとは異なって組み合されてよい、または存在してよい。多様な実施形態では、図８の方法は分散型データベース最適化ストレージシステムの１台または複数のコンピュータによって実行されてよく、その内の１つは図１０のコンピュータシステムとして示される。図８の方法は、システム全体のチェックポイント回避のための方法の１つの例の実装として示される。他の実装では、図８の方法は追加のブロック、または図示されるよりも少ないブロックを含んでよい。

【００９５】

８１０に示されるように、データベースのために記憶される特定のデータページにリンクされるリドゥログレコードが維持されてよい。これらのリドゥログレコード（上述されたように、ＵＬＲと呼ばれることがある）は、ユーザーデータに対する変更を記述してよい。リドゥログレコードは、データページ等のユーザーデータの特定の部分にリンクされてよい。例えば、いくつかの実施形態では、リドゥログレコードは、特定のデータページに最終的にリンクされるリドゥログレコードの連鎖を形成し、各リドゥログレコードはデータページのための以前に受信されたリドゥログレコードを指す。この例を使用すると、３つのリドゥログレコードが特定のデータページにリンクされる場合には、最も最近に受信されたリドゥログレコードは次に最も最近に受信されたリドゥログレコードを指し、次に最も最近に受信されたリドゥログレコードは同様に３番目に最も最近に受信されたリドゥログレコードを指し、３番目に最も最近に受信されたリドゥログレコードはデータページの最も最近に保存された状態を指す。前のリドゥログレコードに対する各ポインタによって示されるリドゥログレコードの論理的な順序付けが、係るリドゥログレコードが係る順序で物理的に記憶されることを暗示しないことに留意されたい。図６に関して上述されたように、これらのリドゥログレコードは、いくつかの実施形態では、ユーザーデータの他の部分にリンクされた他のリドゥログレコードとインタリーブされてよい。したがって、前の例は制限的となることを目的としていない。

【００９６】

多様な実施形態では、リドゥログレコードは、ストレージノード４３０、４４０、４５０等のストレージノードにデータが記憶されてよい、１つまたは複数のデータベースを管理してよいデータベースエンジンヘッドノード４２０等のデータベースシステムから受信されてよい。しかしながら、少なくともいくつかの実施形態では、ストレージノードは、ストレージノードがデータを記憶するための１つまたは複数の追加のデータベースシステムまたはノードからリドゥログレコードを受信してよい。これらの他のデータベースシステムまたはノードは、ストレージノードにそのそれぞれのデータベースのために記憶されているデータの特定の部分にリンクされたリドゥログレコードを送信してもよい。

【００９７】

いくつかの実施形態では、受信されたリドゥログレコードが次いで記憶されてよい。図６は、係るリドゥログレコードがどのようにして受信され、処理され、ストレージノードに記憶されてよいのかの多様な実施形態を説明する。多様な形式のメタデータが、データページ等の特定の部分データにリンクされるリドゥログレコードの数つまりカウント等の記憶されているリドゥログレコードのために維持されてよい。例えば、上記に示された例でのように、３つのリドゥログレコードが特定のデータページにリンクされる場合、次いで特定のデータページのリドゥログレコードカウントは３で維持されてよい。多様な他のログレコードに対するポインタまたはデータページの最も最近に保存された状態に対するポインタ等、サイズまたは物理的な場所、及びリドゥログレコードがリンクされるデータの部分等のリドゥログレコードに関する他のメタデータが維持されてよい。

【００９８】

記憶されているリドゥログレコードのために維持されるメタデータに対する更新は、リドゥログレコード自体に対する変更、それらがリンクされる特定のデータページに対する変更、またはリドゥログレコードを活用することによって、またはリドゥログレコードに関して実行される動作もしくは他の方法もしくは技法に応えて行われてよい。例えば、８３０で示されるように、合体動作が実行され、データページの現在の状態を生成するため

に特定のデータページにリンクされる１つまたは複数のリドゥログレコードを適用する場合、次いでリドゥログレコードカウントは特定のデータページに対するリドゥログレコードカウントからそれらの適用されたリドゥログレコードを削除するために更新されてよい。

【００９９】

多様な実施形態では、特定のデータページのための合体イベントは、特定のデータページにリンクされる１つまたは複数のリドゥログレコードに少なくとも部分的に基づいて、８２０で示されるように検出されてよい。検出された合体イベントは、合体動作が特定のデータページに対して実行されてよいことを示してよい。少なくともいくつかの実施形態では、特定のデータページのための合体イベントを検出することは、他のデータページについて検出された合体イベントとは関係なく、または他のデータページについて検出された合体イベントを考慮せずに発生してよい。特定のデータページが、多くのリドゥログレコードが受信される「ホット」データページであってよいシナリオを考える。リドゥログレコードはめったに他のデータページのために受信されることはない。合体イベントを検出することは、合体閾値を超えるそれぞれのデータページにリンクされるリドゥログレコードの数に基づいてよく、したがって、このシナリオでは、合体イベントは他のデータページについてより、特定の「ホット」データページについてより頻繁に検出されてよい。

【０１００】

合体イベントを検出することは、バックグラウンドプロセスとして実行してよいストレージノード監視構成要素またはプロセスの一部として実行されてよく、読取り要求、書込み要求、及び他のアクセス要求を処理するフォアグラウンドプロセスは、合体イベントの検出の前に（または合体イベントの検出を遅延させて）実行されてよい。合体イベントの検出は、ストレージノードの作業負荷が作業負荷閾値未満であるとき等、周期的な間隔または非周期的な間隔で発生してよい。

【０１０１】

特定のデータページにリンクされたリドゥログレコードに少なくとも部分的に基づいて合体イベントを検出するための多様な方法及び技法が実装されてよい。例えば、少なくともいくつかの実施形態では、合体閾値は合体イベントを検出するために活用されてよい。合体閾値は、合体イベントが検出される前に特定のデータページにリンクされてよいリドゥログレコードの数を定義してよい。例えば、特定のデータページが、１０リドゥログレコードの合体閾値を超える１１リドゥログレコードを有する場合、次いで合体イベントが検出されてよい。異なる合体閾値は異なるデータページに活用されてよい。例えば、データページにリンクされた頻繁なリドゥログレコードを受信する「ホット」データページのシナリオを再度考える。リドゥログレコードをあまり頻繁に受信しないデータページよりも高い合体閾値は、「ホット」データページに活用され、このようにして「ホット」データページに対して実行される合体動作の数を削減してよい。代わりに、いくつかの実施形態では、同じ合体閾値または類似する合体閾値が活用されてよい。合体閾値は、多様な他の技法または構成部品と結合されてもよい。例えば、他の構成部品を使用していつ合体閾値が超えられる可能性があるのかを計算し、タイマまたは他の構成要素を設定して、合体イベント検出を実行するバックグラウンドモニタまたは他のプロセスに対し、特定のデータページのリドゥログレコードカウントが調べられるべきであることを示すこと。

【０１０２】

少なくともいくつかの実施形態では、特定のデータページに対する（またはデータページの特定のセットに対する）合体閾値が決定されてよい。例えば、いくつかの実施形態では、合体閾値はユーザー定義の合体閾値に従って決定されてよい。ユーザー定義の合体閾値は、要求され、決定され、もしくはデータベースエンジンヘッドノード４２０等のデータベースシステムからストレージノードに対して示される合体閾値であってよい、またはデータベースシステムのクライアントは合体イベントを検出するために使用される合体閾値を与えてよい。いくつかの実施形態では、合体閾値はストレージノードの作業負荷または性能に基づいて決定されてよい。例えば、いくつかの実施形態では、作業負荷測度また

は性能測度が、合体動作を実行するための能力が低いことを示す場合、次いで合体閾値は、検出される合体イベントの数がストレージノードによってその現在の作業負荷で処理され得るように増加されてよい。いくつかの実施形態では、リドゥログレコードが特定のデータページについて受信されるレートつまり頻度が計算され、合体閾値を決定するために使用されてよい。少なくともいくつかの実施形態では、リドゥログレコードのサイズ、物理記憶でのリドゥログレコードの場所、リドゥログレコードを記憶するために利用可能なスペース、及び／または合体動作がデータページの以前に記憶されたバージョンにリドゥログレコードを適用するために実行されてよい時刻等の多様な他の特徴が合体閾値を決定するために使用されてよい。

【０１０３】

特定のデータページに対する合体イベントを検出することに応えて、特定のデータページにリンクされる１つまたは複数のリドゥログレコードが、８３０で示されるように特定のデータページをその現在の状態で生成するために特定のデータの以前に記憶されたバージョンに適用されてよい。少なくともいくつかの実施形態では、特定のデータページにリンクされるリドゥログレコードを適用することは合体動作の一部として実行される。上述されたような合体動作つまり合体は、ユーザーページのより最近のバージョンを作成するためにＤＵＬＲ等のリドゥログレコードをユーザーページの初期のバージョンに適用してよい。いくつかの実施形態では、合体動作は、最も最近のＡＵＬＲ（例えば、データページの以前に記憶されたバージョン）の位置を突き止め、ＤＵＬＲのいずれも省略することなくあらゆる以後のＤＵＬＲを順に適用することを含んでよい。例えば、３つのＤＵＬＲが受信され、ＡＵＬＲにリンクされている場合、最初に受信されたＤＵＬＲがＡＵＬＲに適用される（このようにして、以前に記憶されたデータページを基準にして最初に受信された変更を適用する）。次いで、次に受信されたＤＵＬＲが適用され、最後に最も最近のＤＵＬＲが適用され、記憶ノードでのＤＵＬＲの受信に基づいて決定される順にＤＵＬＲを適用する。いくつかの実施形態では、新しいＡＵＬＲは特定のデータページの現在の状態として生成される。リドゥログレコードカウント等の上述されたメタデータは、リドゥログレコードの適用を反映し、リドゥログレコードカウントに関して、その数をカウントから削除するために更新されてよい。

【０１０４】

少なくともいくつかの実施形態では、遅延は、８２０で示される合体イベントの検出と８３０で示されるリドゥログレコードの適用との間で発生してよい、または実行されてよい。例えば、該検出及び該適用を実行するストレージノードの作業負荷が、リドゥログレコードを適用することの実行と、合体イベントの検出との間の遅延を決定してよい。同様に、合体イベントの検出に応えるリドゥログレコードの適用はバックグラウンドプロセスの一部として実行されてよい、すなわち削減される、つまり多様なアクセス要求（例えば、読取り要求または書込み要求）の処理等、フォアグラウンドプロセスを実行しないときにだけ実行される。遅延した合体動作またはデータページのためのリドゥログの適用は、データページがいつリドゥログレコードを適用させるべきであるのかの順序、シーケンス、またはタイミングを決定する、先入先出し（ＦＩＦＯ）待ち行列または優先順位待ち行列等のデータ構造に入れられてよい。例えば、上述されたシナリオでのように、「ホット」データページが検出された合体イベントを有する場合、別のデータページの代わりに「ホット」データページに対するリドゥログの適用を実行する方がより効率的であることがある。バックグラウンドプロセスとしてリドゥログレコードの適用を遅延するまたは実行する結果として、合体イベントが検出されたデータページにリンクされる１つまたは複数の追加のリドゥログレコードが受信されてよい。少なくともいくつかの実施形態では、これらの追加のリドゥログレコードは、他のリドゥログレコードがデータページの以前に記憶されたバージョンに適用されるときに適用されてよい。

【０１０５】

図４に示されるように、複数のストレージノード４３０、４４０、４５０他は、分散型ストレージサービスの一部として実装されてよい。図８に関して上述された多様な方法及

び技法は、これらの複数のストレージノードによって互いと無関係に実行されてよい。各ストレージノードは、合体イベントを検出すること、及びそれに応じて同時にまたは互いと異なるときに１つまたは複数のリドゥログレコードを適用することを実行するだけではなく、異なる合体閾値または同じ合体閾値を決定してもよい。

【０１０６】

ここで、いくつかの実施形態に従って、分散型データベースシステムのための高速クラッシュ回復を実行するための方法を明示する一連の図を示す図９Ａを参照する。典型的なデータベースシステムにおけるクラッシュ回復は達成が困難なプロセスである。これらの典型的なシステムでは、データベースシステム故障からの回復時、データベースのクリーンなバージョンが得られ、次いでディスクに記憶されていないトランザクションからのリドゥログレコードのすべてが、データベースをデータベースシステム故障の前のその現在の状態に復元するためにリプレイされなければならず、データベースにアクセスできるようになる前に多大な復元時間を生じさせる。図９Ａは、一方、クラッシュ回復を実行するためのより高速且つより効率的な技法を提供してよい分散型データベースシステム用の高速クラッシュ回復の説明を提供する。

【０１０７】

シーン９９２で、図２に関して上述されたデータベースクライアント２５０等のデータベースクライアント９０６は、図２に上述されたネットワーク２６０上で、データベースを実装する、図４に関して上述されたデータベースヘッドノード４３０等のデータベースヘッドノード９０２と通信する。ストレージノード９０８は、データベースヘッドノード９０２によって実装されるデータベースのためのログ構造化データストレージを実装する１つまたは複数のストレージノードであってよい。多様なアクセス要求が受信され、その後ストレージノード９０８からアクセスされたデータを取り出すと、データベースヘッドノード９０２によってサービスを提供されてよい。図８に関して上述されたもの等のリドゥログレコードが生成され、ユーザーデータを送信する代わりにストレージノード９０８に送信されてよい。リドゥログレコードはストレージノード９０８で維持されてよい。少なくともいくつかの実施形態では、合体動作は、図８に関して上述されたように等、合体イベントの検出に応えて実行されてよい。

【０１０８】

シーン９９４は、データベースヘッドノード９０２の故障を示す。データベースヘッドノード故障は、電源喪失、利用可能なメモリなし、システム障害等の、データベースヘッドノードが機能を続行できないようにさせる任意のタイプのシステム故障であることがある。データベースクライアント９０６とデータベースヘッドノード９０２との間の通信は、図に示されるように送信または受信されないことがある。したがって、データベースに対するアクセスは提供され得ない。同様に、ストレージノード９０８とデータベースヘッドノード９０２との間の通信が送信または受信されないことがあり、したがってデータベースのために記憶されているデータに対する要求が処理されないことがある。

【０１０９】

シーン９９６では、回復動作が示されてよい。同じシステムハードウェアで再起動されたヘッドノードアプリケーションプログラムのバージョン、または異なるハードウェアで起動されたヘッドノードの別のインスタンスであってよい新しいデータベースヘッドノード９０４がオンラインにされてよい。ストレージノード９０８との接続は、示されるように、データベースヘッドノード９０４によって確立されてよい。シーン９９８は、ストレージノード９０８との接続の確立時、データベースヘッドノード９０２で実装されたのと同じデータベースが、新しいデータベースヘッドノード９０４でのアクセスのために利用可能にされてよいことを示す。読取り要求または書込み要求等のアクセス要求は、ネットワーク２６０を介してデータベースクライアント９０６から新しいデータベースヘッドノード９０４に送信されてよい。リドゥログレコードはすでに、アクセス要求にサービスを提供するために新しいデータベースヘッドノード９０８にデータベースのために記憶されているデータのカレントバージョンを提供してよいストレージノード９０８に送信されて

いたので、新しいデータベースヘッドノード９０４は、データベースヘッドノード故障の前にデータの現在の状態を入手するためにこれらのリドゥログレコードをリプレイする必要がないことがある。ストレージノード９０８は、特定のデータに対する要求が受信されるとき特定のデータの以前に記憶されていたバージョンにリドゥログレコードを適用してよい。代わりに、特定のデータの現在の状態は、図８に関して上述されたように合体イベントが検出されるとき等、あらゆるリドゥログレコードがすでに適用されている特定のデータに向けられた状態でストレージノードにすでに記憶されていてよい。

【０１１０】

図９Ｂは、いくつかの実施形態に係る、分散型データベースシステムのための高速クラッシュ回復を実行する方法を示す流れ図である。多様な実施形態では、データベースヘッドノード故障が発生することがある。このヘッドノード故障はあらゆる通信、修正、または故障したデータベースヘッドノードによって実装され、管理されるデータベースへの他の形のアクセスを妨げることがある。例えば、図２に説明されるデータベースクライアント２５０等のデータベースシステムクライアントは、故障したデータベースヘッドノードに読取り要求または書込み要求を送信できないことがある。データベースヘッドノードの故障は、例えば図２に上述されたウェブサービスプラットホーム２００、または何らかの他のシステムもしくは構成要素によって検出されてよい。ヘッドノードの故障に応えて、再起動されたデータベースヘッドノードまたは新しいデータベースヘッドノード（例えば、以前に故障したヘッドノードと同じまたは異なるハードウェア上でホストされる新しいデータベースヘッドノード仮想インスタンス）が、回復動作を実行するように命令されてよい。いくつかの実施形態では、この回復動作はこれらの要素に制限されていないが、回復動作は図９Ｂに示される多様な要素を含んでよい。

【０１１１】

データベースヘッドノード故障からの回復は、９１０に示されるように発生してよい。回復は実行され、さまざまな方法で完了していると決定されてよい。例えば、データベースヘッドノードアプリケーションは、多様なテストを実行すること、多様な装置を有効にすること等、実行するために準備するときに多様な状態を有することがある。このプロセスの一部として、ノード故障からの回復の完了を示してよいデータベースヘッドノードについて準備完了した状態が決定されてよい。９１０に示されるように、データベースノード故障からの回復時、９２０に示されるように、データベースのためにデータを記憶する１台または複数のストレージノードとの接続が確立されてよい。

【０１１２】

図９Ａ及び上記の多様な他の図に関して上述されたように、データベースは、図３及び図４に説明されるデータベースヘッドノード３２０または４４０等のデータベースヘッドノードによって実装され、管理されてよい。上述された読取り要求または書込み要求等のデータベースアクセス要求を実装することの一部として、データベースヘッドノードで処理されてよい。少なくともいくつかの実施形態では、データベースに対する変更を反映するリドゥログレコードは、ストレージノードに記憶されるデータに対する変更を反映する、図４で上述されたストレージノード４５０等の１つまたは複数のストレージノードに送信される。特定のデータページまたはデータの他の部分等の、変更されるデータを記憶するストレージノードは、変更される、データページ等のデータの部分にリンクされるリドゥログレコードを受信してよい。これらのリドゥログレコードは、データページのカレントバージョンに対する要求に応えて、または合体イベントの検出に応えて等、なんらかの他の時に、データページ等のデータの部分の以前に記憶されていたバージョンに適用されてよい（例えば、合体動作）。データベースのためのリドゥログレコードは、上述された多様な方法で、データベースヘッドノードで実装されるデータベースのために維持されるので、いくつかの実施形態では、ストレージノードはデータベースヘッドノードに、データベースヘッドノード故障の時刻まで最新であると保証されるデータの現在の状態を送信してよい。

【０１１３】

接続の確立先のストレージノードが識別されてよい。例えば、図４で上述されたクライアント側ストレージサービスドライバ４２５は、どのストレージノードがデータベースのためにデータを記憶するのか、及びデータベースのどの部分がストレージノードに記憶されるのかを示す情報を維持してよい。接続要求、または何らかの他の通信メッセージは、図４に関して上述された多様な通信方法の１つを使用して送信されてよい。同様に、肯定応答、及びストレージノード及び／またはデータベースヘッドノードのステータスについての他の情報が交換されてよい。

【０１１４】

９２０に示されるように、１つまたは複数のストレージノードとの接続の確立時、データベースは、９３０に示されるように、アクセスのために利用可能にされてよい。いくつかの実施形態では、アクセスは１つまたは複数のアクセス要求（例えば、読取り要求、書込み要求）に提供されてよい。データベースの可用性の表示が生成され、クライアントに送信されてよい。例えば、データベースがアクセスに利用可能である旨のメッセージがデータベースクライアントに送信されてよい。係るメッセージは、図２に説明されるウェブサービスプラットホーム２００、またはなんらかの他の通信プラットホームもしくは装置を介して送信されてよい。上述されたように、典型的なデータベースシステムでは、リドゥログレコードのリプレイは、データベースを利用可能にする前に実行されなければならない。しかし、少なくともいくつかの実施形態では、データベースはリドゥログレコードをリプレイせずに利用可能にされてよい。リドゥログレコードとともに使用されるときの用語「リプレイ」が概してデータの以前に記憶されていたバージョンに対して１つまたは複数のリドゥログレコードを適用することを意味することに留意されたい。

【０１１５】

少なくともいくつかの実施形態では、ストレージノードは、データベースヘッドノード故障を検出できてよい、またはそれ以外の場合データベースヘッドノード故障を認識させられてよい。データベースヘッドノード故障の検出に応えて、ストレージノードは、ストレージノードで受信されたリドゥログレコードに対する切り詰め演算を実行してよい。切り詰め演算は、データベースヘッドノードの故障の前に完了しなかったシステムトランザクションの一部であるリドゥログレコードを決定してよい、または識別してよい。これらの識別されたリドゥログレコードは、それらがリンクされているデータページにそれらが適用され得ないように、削除されてよい、またはそれ以外の場合、マークされてよい、移動されてよい、もしくは識別されてよい。例えば、記憶ページが特定のデータページのために５リドゥログレコードを維持し、最も最近の３リドゥログレコードが、データベースヘッドノード故障の前に完了しなかったシステムトランザクションの一部である場合、次いでストレージノードは、２つの最も旧いリドゥログレコードだけを適用することによってデータページの現在の状態を生成するときに最も最近の３リドゥログレコードを無視してよい。少なくともいくつかの実施形態では、切り詰め演算は、回復されたデータベースヘッドノードと接続を確立できるようになる前に、影響を受けたリドゥログレコードがあるストレージノードに対して実行されてよい。データベースエンジンヘッドノードは、いくつかの実施形態では、データベースヘッドノードの故障前に完了しなかったシステムトランザクションの一部であるリドゥログレコードを同様に決定し、または識別し、これらの識別されたリドゥログレコードが、それらがリンクされているデータページにそれらが適用され得ないように削除されてよい、またはそれ以外の場合マークされてよい、移動されてよい、または識別されてよい旨の通知をストレージノードに送信するように構成されてよい。例えば、図３に関して上述されたクライアント側ストレージサービスドライバ３２５等のクライアント側ストレージサービスドライバは、上述された技法を実行してよい。切り詰め演算を説明するこれらの技法は、いくつかの実施形態では、バックグラウンドプロセスの一部として実行されてよい。

【０１１６】

少なくともいくつかの実施形態では、システムトランザクションは、ユーザートランザクションを実行する、または実装するための動作または他の形の１つもしくは複数のタス

クであってよい。ユーザートランザクションは、受信されたアクセス要求から多様なタスクまたは動作を実行するために複数のシステムトランザクションを含んでよい。例えば、データベースに対する挿入命令が受信されてよい。ユーザートランザクションとして、この挿入命令は、挿入を実行するために、例えばｂ－ツリー等のデータベースデータ構造に作用する等、挿入を実行するための複数のシステムトランザクションを含んでよい。少なくともいくつかの実施形態では、不完全なユーザートランザクションは、ユーザートランザクションであり、該ユーザートランザクションに含まれるシステムトランザクションのすべてが完了していない（または耐久的にされていない）可能性がある。同様に、システムトランザクションは不完全なことがある。ユーザートランザクション及びシステムトランザクションの一部としてデータベースのために記憶されたデータに対して行われた変更を反映するリドゥログレコードは、いくつかの実施形態では、特定のユーザートランザクション及び／またはシステムトランザクションで識別されてよい。

【０１１７】

図９Ｃは、いくつかの実施形態に係る、回復されたデータベースでアクセス要求を処理するための方法を示す流れ図である。上述されたように、少なくともいくつかの実施形態では、アクセスのためにデータベースを利用できるようにしたデータベースヘッドノードで、アクセス要求が受信されてよい。アクセス要求は、読取り要求、書込み要求、またはデータベースのために記憶されているデータを入手するもしくは修正するための任意の他の要求であってよい。図９Ｃが示すように、アクセス要求は、９４０で示されるようにデータベースに対して受信されてよい。それに応じて、９５０示されるように１つまたは複数のストレージノードからの１つまたは複数のデータページに対する要求が行われてよい（クライアントからのアクセス要求及びデータベースヘッドノードからのデータ要求の両方とも、上記図５に関してより詳細に扱われている）。要求された１つまたは複数のデータページの現在の状態は、９６０に示されるように、ストレージノードから受信されてよい。上述されたように、この現在の状態は、データページの以前に記憶されたバージョンまで以前に受信されたリドゥログレコードをリプレイする、もしくはデータページの以前に記憶されたバージョンに以前に受信されたリドゥログレコードを適用することによって、または現在の状態であるデータページの以前に記憶されたバージョンを返すことによって生成されてよい。多様な実施形態では、各データページまたは要求されたデータの一部は、（例えばゆったりと）データに対する要求を受信することに応えて、その現在の状態を決定させる、生成させる、及び／または送り返させてよい。

【０１１８】

少なくともいくつかの実施形態では、アンドゥログレコードは、データベースヘッドノードで維持されてよい。上述されたようなアンドゥログレコードは、不完全なユーザートランザクションが発生した場合に等、データに対して行われた変更をアンドゥするためにデータベースのために記憶されるデータに適用される変更を記録してよい。ユーザートランザクションは、（複数のシステムトランザクション等の）データベースのために記憶されるデータに対する複数の変更を含み、１つまたは複数のリドゥログレコード及び１つまたは複数のアンドゥログレコードを生成してよい。ユーザートランザクションは、ユーザートランザクションの変更のすべてがコミットされなかった（例えば、耐久的にされなかった）ときに不完全であることがある。図３に関して上述されたトランザクションログ３４０等のトランザクションテーブルは、どのユーザートランザクション、及びストレージノードに記憶されているデータのその関連付けられた部分が、データベースヘッドノード故障前にコミットされず、したがって不完全であるのかを示すために実装されてよい。９７０で示されるように、受信されたデータページがトランザクションテーブルによって示される等、不完全なユーザートランザクションによって影響を及ぼされるかどうかに関して決定が下されてよい。はいである場合、肯定の出口が示すように、次いでアンドゥログレコードの１つまたは複数が、不完全なトランザクションによって行われた変更をアンドゥして、９７２に示すように、データページの新しい現在の状態を生成するためにデータページに適用されてよい。アンドゥログレコードが適用された、つまり不完全なユーザー

トランザクションによってデータページが影響を及ぼされなかった後、次いでデータページの現在の状態が、９８０で示されるようにアクセス要求にサービスを提供するために提供されてよい。

【０１１９】

少なくともいくつかの実施形態で、トランザクションテーブルに基づいて、不完全なユーザートランザクションによって影響を受けたデータの部分を決定する、または識別するバックグラウンドプロセスが実行されてよい。不完全なユーザートランザクションによって影響を受けた、データページ等のデータの部分の現在の状態に対する要求が送受されてよい。アンドゥログレコードは、次いで、不完全なユーザートランザクションによってこれらのデータページに向けられた変更をアンドゥするために適用されてよい。多様な実施形態では、データベースキャッシュが、アンドゥログレコードが適用された後にこれらのデータページで更新されてよい。

【０１２０】

少なくともいくつかの実施形態では、以前に記録されたスナップショットが、データベースの状態を初期の状態に復元するために使用されてよい。例えば、アクセスのためにデータベースを利用可能にする前に、要求は、データベースのためのデータを以前に記録されたスナップショットに対応する状態に復元するためにストレージノードに送信されてよい。スナップショットは、以前に受信されたリドゥログレコードを、記録されたスナップショット点（例えば、タイムスタンプまたはマーカ）までリプレイできるようにする、ストレージノードに記憶されるリドゥログのためのタイムスタンプまたは他のマーカまたはインジケータを識別することによって記録されてよく、該復元は複数のリドゥログの１つまたは複数をデータの以前のバージョンに適用することを含む。ストレージノードにスナップショットを実装する追加説明が上記に示される。

【０１２１】

図９Ｂから図９Ｃの方法及び技法は、データベースエンジンヘッドノード４２０等のデータベースシステムの多様な構成要素によって実行されるとして説明されてよいが、方法は、いくつかの場合、いずれの特定の構成要素によっても実行される必要はない。例えば、いくつかの場合、図９Ｂから図９Ｃは、いくつかの実施形態に従って、なんらかの他の構成要素またはコンピュータシステムによって実行されてよい。また、いくつかの場合、データベースシステム４００の構成部品は、データベースシステム４００の構成要素は、図４の例に示されるのとは異なって組み合されてよい、または存在してよい。多様な実施形態では、図９Ｂから図９Ｃの方法は分散型データベースシステムの１台または複数のコンピュータによって実行されてよく、その内の１つは図１０のコンピュータシステムとして示される。図９Ｂから図９Ｃの方法は、分散型データベースシステムの高速クラッシュ回復のための方法の例の実装として示される。他の実装では、図９Ｂから図９Ｃの方法は追加のブロック、または図示されるよりも少ないブロックを含んでよい。

【０１２２】

本明細書に説明される方法は、多様な実施形態では、ハードウェア及びソフトウェアの任意の組合せによって実装されてよい。例えば、一実施形態では、方法は、プロセッサに結合されたコンピュータ可読記憶媒体に記憶されるプログラム命令を実行する１台または複数のプロセッサを含むコンピュータシステム（例えば、図１０のコンピュータシステム）によって実装されてよい。プログラム命令は、本明細書に説明される機能性（例えば、本明細書に説明されるデータベースサービス／システム及び／またはストレージサービス／システムを実装する多様なサーバ及び他の構成要素の機能性）を実装するように構成されてよい。

【０１２３】

図１０は、多様な実施形態に従って、本明細書に説明されるデータベースシステムの少なくとも一部を実装するように構成される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１０００は、異なる実施形態で、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、またはデータベース階層のクライアントの代わりにデ

ータベース及び関連付けられたメタデータを記憶する別個の分散型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内の１つを実装するように構成されてよい。コンピュータシステム１０００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またはノートパソコン、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、消費者装置、アプリケーションサーバ、ストレージデバイス、電話、携帯電話、または一般的に任意のタイプのコンピューティング装置を含むが、これに限定されることがない多様なタイプの装置のいずれかであってよい。

【０１２４】

コンピュータシステム１０００は、入出力（Ｉ／Ｏ）インタフェース１０３０を介してシステムメモリ１０２０に結合される（いずれかが、単一スレッドまたはマルチスレッドであってよい複数のコアを含んでよい）１台または複数のプロセッサ１０１０を含む。コンピュータシステム１０００は、Ｉ／Ｏインタフェース１０３０に結合されるネットワークインタフェース１０４０をさらに含む。多様な実施形態では、コンピュータシステム１０００は、１台のプロセッサ１０１０を含んだユニプロセッサシステム、または数台のプロセッサ１０１０（例えば、２，４、８、または別の適切な数）を含んだマルチプロセッサシステムであってよい。プロセッサ１０１０は、命令を実行できる任意の適切なプロセッサであってよい。例えば、多様な実施形態では、プロセッサ１０１０は、ｘ８６、ＰｏｗｅｒＰＣ、ＳＰＡＲＣ、もしくはＭＩＰＳ　ＩＳＡ等のさまざまな命令セットアーキテクチャ（ＩＳＡ）または任意の他の適切な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１０１０のそれぞれが、一般に同じＩＳＡを実装してよいが、必ずしも同じＩＳＡを実装しないこともある。コンピュータシステム１０００は、通信ネットワーク（例えば、インターネット、ＬＡＮ等）上で他のシステム及び／または構成要素と通信するための１台または複数のネットワーク通信装置（例えば、ネットワークインタフェース１０４０）も含む。例えば、システム１０００で実行中のクライアントアプリケーションは、単一のサーバ上、または本明細書で説明されるデータベースシステムの構成要素の内の１つまたは複数の実装するサーバのクラスタ上で実行中のサーバアプリケーションと通信するためにネットワークインタフェース１０４０を使用してよい。別の例では、コンピュータシステム１０００上で実行中のサーバアプリケーションのインスタンスは、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１０９０）の上で実装されてよい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するために、ネットワークインタフェース１０４０を使用してよい。

【０１２５】

示されている実施形態では、コンピュータシステム１０００は、１台または複数の永続ストレージデバイス１０６０及び／または１台または複数のＩ／Ｏデバイス１０８０も含む。多様な実施形態では、永続ストレージデバイス１０６０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、他の大容量記憶装置、または任意の他の永続ストレージデバイスに相当してよい。コンピュータシステム１０００（または、コンピュータシステム１０００上で動作する分散アプリケーションもしくはオペレーティングシステム）は、所望されるように、命令及び／またはデータを永続ストレージデバイス１０６０に記憶してよく、必要に応じて記憶されている命令及び／またはデータを取り出してよい。例えば、いくつかの実施形態では、コンピュータシステム１０００は、ストレージシステムサーバノードをホストしてよく、永続記憶装置１０６０はそのサーバノードにアタッチされるＳＳＤを含んでよい。

【０１２６】

コンピュータシステム１０００は、プロセッサ（複数の場合がある）１０１０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る１つまたは複数のシステムメモリ１０２０を含む。多様な実施形態では、システムメモリ１０２０は、任意の適切なメモリ技術（例えば、キャッシュ、スタティックランダムアクセスメモリ（ＳＲＡＭ）、Ｄ

ＲＡＭ、ＲＤＲＡＭ、ＥＤＯ　ＲＡＭ、ＤＤＲ　１０ＲＡＭ、同期ダイナミックＲＡＭ（ＳＤＲＡＭ）、Ｒａｍｂｕｓ　ＲＡＭ、ＥＥＰＲＯＭ、不揮発性／フラッシュタイプメモリ、または任意の他のタイプのメモリの内の１つまたは複数）を使用して実装されてよい。システムメモリ１０２０は、本明細書に説明される方法及び技法を実装するためにプロセッサ（複数の場合がある）１０１０によって実行可能であるプログラム命令１０２５を含んでよい。多様な実施形態では、プログラム命令１０２５は、プラットホームネイティブバイナリ、Ｊａｖａ（商標）バイトコード等の任意のインタープリター型言語で、またはＣ／Ｃ＋＋、Ｊａｖａ（商標）等の任意の他の言語で、またはその任意の組合せで符号化されてよい。例えば、示されている実施形態では、プログラム命令１０２５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、または異なる実施形態で、データ階層のクライアントの代わりにデータベース及び関連付けられたメタデータを記憶する別個の分散型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内の１つの機能性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１０２５は、複数の別個のクライアント、サーバノード、及び／または他の構成要素を実装してよい。

【０１２７】

いくつかの実施形態では、プログラム命令１０２５が、ＵＮＩＸ（登録商標）、ＬＩＮＵＸ、Ｓｏｌａｒｉｓ（商標）、ＭａｃＯＳ（商標）、Ｗｉｎｄｏｗｓ（商標）等の多様なオペレーティングシステムの内のいずれかであってよいオペレーティングシステム（不図示）を実装するために実行可能な命令を含んでよい。プログラム命令１０２５のいずれかまたはすべては、多様な実施形態に従ってプロセスを実行するためにコンピュータシステム（または他の電子機器）をプログラミングするために使用されてよい、その上に記憶されている命令を有する非一過性のコンピュータ可読記憶媒体を含んでよいコンピュータプログラム製品、つまりソフトウェアとして提供されてよい。非一過性の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）によって読取り可能な形（例えば、ソフトウェア、処理アプリケーション）をとる情報を記憶するための任意の機構を含んでよい。一般的に言えば、非一過性のコンピュータアクセス可能記憶媒体は、例えばＩ／Ｏインタフェース１０３０を介してコンピュータシステム１０００に結合される、ディスクまたはＤＶＤ／ＣＤ－ＲＯＭ等の磁気媒体または光学媒体等の、コンピュータ可読記憶媒体または記憶媒体を含んでよい。また、非一過性のコンピュータ可読記憶媒体は、コンピュータシステム１０００のいくつかの実施形態では、システムメモリ１０２０または別のタイプのメモリとして含まれてよい、ＲＡＭ（例えば、ＳＤＲＡＭ、ＤＤＲ　ＳＤＲＡＭ、ＲＤＲＡＭ、ＳＲＡＭ等）、ＲＯＭ等の任意の揮発性媒体または不揮発性媒体を含んでもよい。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フェース１０４０を介して実装されてよい等、ネットワークリンク及び／または無線リンク等の通信媒体を介して伝達される、光信号、音響信号、または他の形の伝搬信号（例えば、搬送波、赤外線信号、デジタル信号等）を使用して通信されてよい。

【０１２８】

いくつかの実施形態では、システムメモリ１０２０は、本明細書に説明されるように構成されてよいデータストア１０４５を含んでよい。例えば、本明細書に説明されるデータベース階層の機能を実行する際に使用されるトランザクションログ、アンドゥログ、キャッシュに入れられたページデータ、または他の情報等の、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）記憶されるとして本明細書に説明される情報は、データストア１０４５にもしくは１つまたは複数のノード上のシステムメモリ１０２０の別の部分に、永続記憶装置１０６０に、及び／または１つまたは複数のリモートストレージデバイス１０７０に異なるときに及び多様な実施形態で記憶されてよい。同様に、記憶階層によって記憶されているとして本明細書に説明される情報（例えば、本明細書に説明される分散型ストレージシステムの機能を実行する上で使用されるリドゥログレコード、合体データページ、及び／または他の情報）は、データストア１０４５にもしくは１つまたは複数のノード上のシステムメモリ１０２０の別の部分に、永続記憶装置１０

６０に、及び／または１つまたは複数のリモートストレージデバイス１０７０に異なるときに及び多様な実施形態で記憶されてよい。一般に、システムメモリ１０２０（例えば、システムメモリ１０２０の中のデータストア１０４５）、永続記憶装置１０６０、及び／またはリモートストレージ１０７０は、データブロック、データブロックのレプリカ、データブロックと関連付けられたメタデータ、及び／またはその状態、データベース構成情報、及び／または本明細書に説明される方法及び技法を実装する上で使用できる任意の他の情報を記憶してよい。

【０１２９】

一実施形態では、Ｉ／Ｏインタフェース１０３０は、プロセッサ１０１０と、システムメモリ１０２０と、ネットワークインタフェース１０４０または他の周辺インタフェースを通してを含んだシステムのあらゆる周辺装置との間のＩ／Ｏトラフィックを調整するように構成されてよい。いくつかの実施形態では、Ｉ／Ｏインタフェース１０３０は、１つの構成要素（例えば、システムメモリ１０２０）から別の構成要素（例えば、プロセッサ１０１０）による使用に適したフォーマットにデータ信号を変換するために任意の必要なプロトコル、タイミング、または他のデータ変形を実行してよい。いくつかの実施形態では、Ｉ／Ｏインタフェース１０３０は、例えばペリフェラルコンポーネントインターコネクト（ＰＣＩ）バス規格、またはユニバーサルシリアルバス（ＵＳＢ）規格の変形等の多様なタイプの周辺バスを通してアタッチされるデバイスに対するサポートを含んでよい。いくつかの実施形態では、Ｉ／Ｏインタフェース１０３０の機能は、例えばノースブリッジ及びサウスブリッジ等、２つ以上の別々の構成要素に分割されてよい。また、いくつかの実施形態では、システムメモリ１０２０へのインタフェース等、Ｉ／Ｏインタフェース１０３０の機能性のいくつかまたはすべては、プロセッサ１０１０の中に直接的に組み込まれてよい。

【０１３０】

ネットワークインタフェース１０４０は、例えば、コンピュータシステム１０００と、（本明細書に説明される１つまたは複数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／またはデータベースシステムのクライアントを実装してよい）他のコンピュータシステム１０９０等の、ネットワークにアタッチされる他のデバイスとの間でデータを交換できるように構成されてよい。さらに、ネットワークインタフェース１０４０は、コンピュータシステム１０００と多様なＩ／Ｏ装置１０５０及び／またはリモートストレージ１０７０との間の通信を可能にするように構成されてよい。入出力装置１０５０は、いくつかの実施形態では、１つまたは複数のディスプレイ端末、キーボード、キーパッド、タッチパッド、スキャン装置、音声認識装置もしくは光学認識装置、または１つまたは複数のコンピュータシステム１０００によってデータを入力するまたは取り出すために適した任意の他の装置を含んでよい。複数の入出力装置１０５０は、コンピュータシステム１０００に存在してよい、またはコンピュータシステム１０００を含む分散型システムの多様なノードで分散されてよい。いくつかの実施形態では、類似する入出力装置はコンピュータシステム１０００とは別個であってよく、ネットワークインタフェース１０４０上で等、有線接続または無線接続を通してコンピュータシステム１０００を含む分散型システムの１つまたは複数のノードと対話してよい。ネットワークインタフェース１０４０は、一般に１つまたは複数の無線ネットワークプロトコル（例えば、Ｗｉ－Ｆｉ／ＩＥＥＥ　８０２．１１、または別の無線ネットワーキング規格）をサポートしてよい。ただし、多様な実施形態では、ネットワークインタフェース１０４０は、例えば他のタイプのイーサネット（登録商標）ネットワーク等、任意の適切な有線汎用データネットワークまたは無線汎用データネットワークを介する通信をサポートしてよい。さらに、ネットワークインタフェース１０４０は、Ｆｉｂｒｅ　Ｃｈａｎｎｅｌ　ＳＡＮ等のストレージエリアネットワークを介して、または任意の他の適切なタイプのネットワーク及び／またはプロトコルを介して、アナログ音声ネットワークまたはデジタルファイバ通信ネットワーク等の電気通信ネットワーク／電話網を介する通信をサポートしてよい。多様な実施形態では、コンピュータシステム１０００は、図１０に示される構成要素より多

い、少ない、または異なる構成要素（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インタフェース、イーサネットインタフェース、フレームリレーインタフェース等の他のネットワークインタフェース等）を含んでよい。

【０１３１】

本明細書に説明される分散型システムの実施形態のいずれも、またはその構成要素のいずれも１つまたは複数のウェブサービスとして実装されてよいことに留意されたい。例えば、データベースシステムのデータベース階層の中のデータベースエンジンヘッドノードは、データベースサービス、及び／または本明細書に説明される分散型ストレージシステムを利用する他のタイプのデータストレージサービスをウェブサービスとしてのクライアントに提示してよい。いくつかの実施形態では、ウェブサービスは、ネットワーク上で相互運用可能なマシン対マシンの対話をサポートするように設計されたソフトウェアシステム及び／またはハードウェアシステムによって実装されてよい。ウェブサービスは、ウェブサービス記述言語（ＷＳＤＬ）等のマシン処理可能なフォーマットで記述されるインタフェースを有してよい。他のシステムは、ウェブサービスのインタフェースの記述によって規定される方法でウェブサービスと対話してよい。例えば、ウェブサービスは、他のシステムが呼び出してよい多様な動作を定義してよく、多様な動作を要求するときに他のシステムが準拠することを期待されてよい特定のアプリケーションプログラミングインタフェース（ＡＰＩ）を定義してよい。

【０１３２】

多様な実施形態では、ウェブサービスは、ウェブサービス要求と関連付けられるパラメータ及び／またはデータを含むメッセージを使用することによって要求されてよい、または呼び出されてよい。係るメッセージは、拡張マークアップ言語（ＸＭＬ）等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よい、及び／またはシンプルオブジェクトアクセスプロトコル（ＳＯＡＰ）等のプロトコルを使用してカプセル化されてよい。ウェブサービス要求を実行するために、ウェブサービスクライアントは、要求を含むメッセージをアセンブルし、ハイパテキスト転送プロトコル（ＨＴＴＰ）等のインターネットベースのアプリケーション層転送プロトコルを使用して、メッセージをウェブサービスに対応するアドレス可能なエンドポイント（例えば、ユニフォームリソースロケータ（ＵＲＬ））に伝達してよい。

【０１３３】

いくつかの実施形態では、ウェブサービスは、メッセージベースの技法よりむしろ、表象状態転送（「ＲＥＳＴｆｕｌ」）技法を使用して実装されてよい。例えば、ＲＥＳＴｆｕｌ技法に従って実装されるウェブサービスは、ＳＯＡＰメッセージの中でカプセル化されるよりむしろ、ＰＵＴ、ＧＥＴ、またはＤＥＬＥＴＥ等のＨＴＴＰ方法の中に含まれるパラメータを通して呼び出されてよい。

【０１３４】

以下の実施形態は以下の節を鑑みてさらによく理解されてよい。

１．分散型ストレージシステムを実装する複数のストレージノードであって、分散型ストレージシステムがデータベースのためにログ構造化データストレージを実装するように構成され、複数のリドゥログレコードが複数のストレージノードでデータベースシステムから以前に受信されたことがあり、リドゥログレコードのそれぞれが複数のストレージノードの中でデータベースのために記憶されるデータに対する変更を記述する、複数のストレージノードと、

データベースシステムを実装するデータベースヘッドノードであって、

複数のストレージノードとの接続を確立する、及び

複数のストレージノードとの接続の確立時に、１つまたは複数のアクセス要求のためのデータベースへのアクセスを提供する

ための故障回復動作を実行するように構成される、データベースヘッドノードと、

を備えるシステム。

２．複数のリドゥログレコードをリプレイすることなく、アクセスがデータベースに提

供される、節１に記載のシステム。
３．データベースシステムヘッドノードが、
データベースに対するアクセス要求を受信する、
受信されたアクセス要求に基づいて、ストレージノードに記憶されるデータページの現在状態に対する要求を複数のストレージノードの内の１つに送信する、及び
複数のリドゥログレコードの１つまたは複数が、ストレージノードでデータページをその現在の状態で生成するためにデータページの以前に保存された状態に適用された、要求されたデータページをその現在の状態でストレージノードから受信する、
ようにさらに構成される、節２に記載のシステム。
４．複数のストレージノードに送信される複数のリドゥログレコードの少なくともいくつかがシステムトランザクションを含み、複数のストレージノードの内の１つのストレージノードが、
システムトランザクションが不完全であると決定する、及び
少なくともいくつかのリドゥログレコードによって変更されるデータページの現在の状態を生成するときに適用されないとして複数のリドゥログレコードの少なくともいくつかを識別する
ように構成される、節１に記載のシステム。
５．複数のストレージノードに送信される複数のリドゥログレコードの少なくともいくつかがシステムトランザクションを含み、データベースシステムヘッドノードが、
システムトランザクションが不完全であると決定する、
少なくともいくつかのリドゥログレコードによって変更されるデータページの現在の状態を生成するときに適用されないとして複数のリドゥログレコードの少なくともいくつかを識別する、及び
複数のストレージノードの１つまたは複数に、適用されない、複数のリドゥログレコードの識別された少なくともいくつかを示す通知を送信する
ようにさらに構成される、節１に記載のシステム。
６．データベースヘッドノードを実装する１台または複数のコンピューティング装置によって、
データベースヘッドノード故障からの回復時に、
データベースのためのデータを記憶する分散型ストレージシステムを実装する複数のストレージノードの１つまたは複数のストレージノードと接続を確立することであって、分散型ストレージシステムが、データベースのためにログ構造化データストレージを実装するように構成され、複数のリドゥログレコードが複数のストレージノードで以前に受信されたことがあり、リドゥログレコードのそれぞれが、データベースのために記憶されたデータに対する変更を、それが受信されたそれぞれのストレージノードで記述する、接続を確立することと、
複数のストレージノードの１つまたは複数のストレージノードとの接続の確立時に、アクセスのためにデータベースを利用可能にすることと、
を実行することを、
含む方法。
７．データベースに対するアクセス要求を受信することと、
アクセス要求を受信することに応えて、１つまたは複数のストレージノードからデータベースのためのデータの部分を記憶する１つまたは複数のデータページの現在の状態を要求することと、
アクセス要求にサービスを提供するためにデータベースのためのデータの部分を記憶する１つまたは複数のストレージノードから１つまたは複数のデータページの現在の状態を受信することと、
をさらに含む、節６に記載の方法。
８．受信されたデータベースのためのデータの部分を記憶する１つまたは複数のデータページの少なくとも１つの現在の状態が、１つまたは複数のストレージノードの１つが、

少なくとも１つのデータページの以前に記憶されたバージョンまで複数のリドゥログレコードの１つまたは複数をリプレイすることによって生成される、節７に記載の方法。
９．受信されたデータベースのためのデータの部分を記憶する１つまたは複数のデータページの少なくとも１つからの異なるデータページの現在の状態が、データページの以前に記憶されたバージョンまで複数のリドゥログレコードの１つまたは複数をリプレイすることなく、１つまたは複数のストレージノードの内の１つによって送信される、節８に記載の方法。
１０．データベースヘッドノードが、複数のストレージノードにリドゥログレコードとして送信される変更をアンドゥするために複数のアンドゥログレコードを維持し、方法が、
１つまたは複数のストレージノードから受信される１つまたは複数のデータページの１つが不完全なユーザートランザクションによって影響を及ぼされると決定することであって、ユーザートランザクションが、１つのデータページを含んだ１つまたは複数のストレージノードに記憶されるデータに対して変更を向ける、決定することと、
ユーザートランザクションによってデータページに向けられた変更をアンドゥするためにデータページに１つまたは複数のアンドゥログレコードを適用することと、
をさらに含む、節７に記載の方法。
１１．データベースヘッドノードが、ユーザートランザクションを含んだ複数の不完全なユーザートランザクションを示すトランザクションテーブルを維持し、方法が、
トランザクションテーブルに少なくとも部分的に基づいて、複数の不完全なユーザートランザクションの少なくとも１つによって影響を及ぼされる１つまたは複数の追加のデータページを決定することと、
１つまたは複数のストレージノードから１つまたは複数の追加のデータページの現在の状態を要求することと、
１つまたは複数の追加のデータページを受信することに応えて、少なくとも１つの不完全なユーザートランザクションによって１つまたは複数の追加のデータページに向かって向けられる変更をアンドゥするために、１つまたは複数の追加のデータページに追加の１つまたは複数のアンドゥログレコードを適用することと、
をさらに含む、節１０に記載の方法。
１２．１つまたは複数の追加のデータページを該決定すること、１つまたは複数の追加のデータページを該要求すること、及び１つまたは複数の追加のデータページに追加の１つまたは複数のアンドゥログレコードを該適用することが、データベースヘッドノードでバックグラウンドプロセスの一部として実行され、アクセス要求を該受信すること、１つまたは複数のデータページの現在の状態を該要求すること、及び１つまたは複数のデータページの現在の状態を該受信することがフォアグラウンドプロセスの一部として実行される、節１１に記載の方法。
１３．データベースヘッドノード故障からの回復時に、
アクセスのためにデータベースを利用可能にする前に、データベースのために複数のストレージノードに記憶されたデータの、以前に記録されたスナップショットに対応する状態への復元に対する要求を複数のストレージノードに送信することであって、該復元が複数のリドゥログの１つまたは複数をデータの以前のバージョンに適用することを含む、送信することと、
をさらに含む、節６に記載の方法。
１４．データベースが複数のリドゥログレコードをリプレイすることなくアクセスのために利用可能にされる、節６に記載の方法。
１５．１台または複数のコンピューティング装置による実行時に、
データベースヘッドノード故障からの回復時に、
データベースのためのデータを記憶する分散型ストレージシステムを実装する複数のストレージノードの１つまたは複数のストレージノードとの接続を確立することであって、分散型ストレージシステムがデータベースのためにログ構造化データストレージを実装す

るように構成され、複数のリドゥログレコードが複数のストレージノードで以前に受信されたことがあり、リドゥログレコードのそれぞれが、データベースのために記憶されたデータに対する変更を、それが受信されたそれぞれのストレージノードで記述する、接続を確立すること、及び
複数のストレージノードの１つまたは複数のストレージノードとの接続の確立時に、１つまたは複数のアクセス要求のためにデータベースへのアクセスを提供すること、
を実装するデータベースシステムのデータベースヘッドノードを実装するプログラム命令を記憶する非一過性のコンピュータ可読記憶媒体。
１６．複数のストレージノードで以前に受信された複数のリドゥログレコードが、該データベースヘッドノードとは異なるデータベースヘッドノードから受信された、節１５に記載の非一過性のコンピュータ可読記憶媒体。
１７．複数のリドゥログレコードをリプレイすることなく、アクセスがデータベースに提供される、節１５に記載の非一過性のコンピュータ可読記憶媒体。
１８．データベースシステムヘッドノードが、
データベースに対するアクセス要求を受信することと、
アクセス要求を受信することに応えて、１つまたは複数のストレージノードからデータベースのためのデータの部分を記憶する１つまたは複数のデータページの現在の状態を要求することと、
アクセス要求にサービスを提供するためにデータベースのためのデータの部分を記憶する１つまたは複数のデータページの現在の状態を受信することであって、１つまたは複数の受信されたデータページの少なくとも１つの現在の状態が、１つまたは複数のストレージノードの１つが、少なくとも１つのデータページの以前に記憶されたバージョンまで複数のリドゥログレコードの１つまたは複数をリプレイすることによって生成される、受信することと、
をさらに実装する、節１５に記載の非一過性のコンピュータ可読記憶媒体。
１９．受信されたアクセス要求が読取り要求または書込み要求である、節１８に記載の非一過性のコンピュータ可読記憶媒体。
２０．データベースヘッドノードが、リドゥログレコードとして複数のストレージノードに送信された変更をアンドゥするために、複数のアンドゥログレコードを維持し、データベースヘッドノードが、
１つまたは複数のストレージノードから受信される１つまたは複数のデータページの１つが不完全なユーザートランザクションによって影響を及ぼされると決定することであって、ユーザートランザクションが１つのデータページを含んだ１つまたは複数のストレージノードに記憶されるデータに対して変更を向ける、決定することと、
ユーザートランザクションによってデータページに向けられた変更をアンドゥするためにデータページに１つまたは複数のアンドゥログレコードを適用することと、
をさらに実装する、節１８に記載の非一過性のコンピュータ可読記憶媒体。
２１．データベースヘッドノードが、ユーザートランザクションを含んだ複数の不完全なユーザートランザクションを示すトランザクションテーブルを維持し、データベースシステムヘッドノードが、
トランザクションテーブルに少なくとも部分的に基づいて、複数の不完全なユーザートランザクションの少なくとも１つによって影響を及ぼされる１つまたは複数の追加のデータページを決定することと、
１つまたは複数のストレージノードから１つまたは複数の追加のデータページの現在の状態を要求することと、
１つまたは複数の追加のデータページを受信することに応えて、少なくとも１つの不完全なユーザートランザクションによって１つまたは複数の追加のデータページに向かって向けられる変更をアンドゥするために、１つまたは複数の追加のデータページに追加の１つまたは複数のアンドゥログレコードを適用することと、
をバックグラウンドプロセスとして実行すること

をさらに実装する、節２０に記載の非一過性のコンピュータ可読記憶媒体。

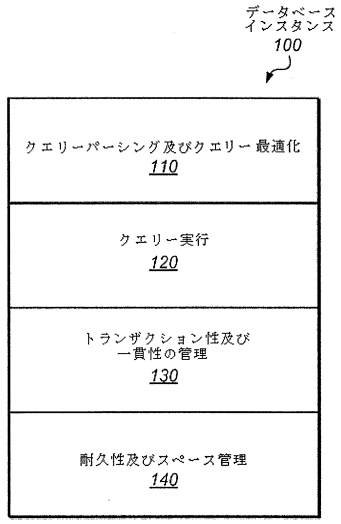
【０１３５】

図に示され、本明細書に説明される多様な方法は、方法の例の実施形態を表す。方法は、ソフトウェアで、ハードウェアで、またはソフトウェア及びハードウェアの組合せで手動で実装されてよい。任意の方法の順序は変更されてよく、多様な要素が追加、再順序付け、結合、省略、修正等、されてよい。

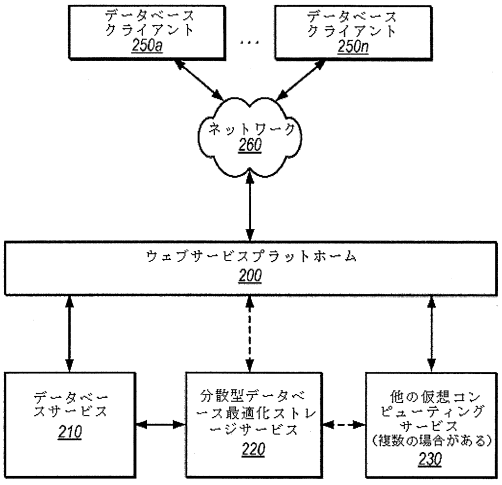
【０１３６】

上記実施形態はかなり詳細に説明されているが、いったん上記開示が完全に理解されると当業者に明らかになるように、多数の変形形態及び修正形態が加えられてよい。続く特許請求の範囲が、すべての係る修正形態及び変更を包含すると解釈され、したがって上記説明は制限的な意味よりむしろ例示的な意味で考えられることが意図される。

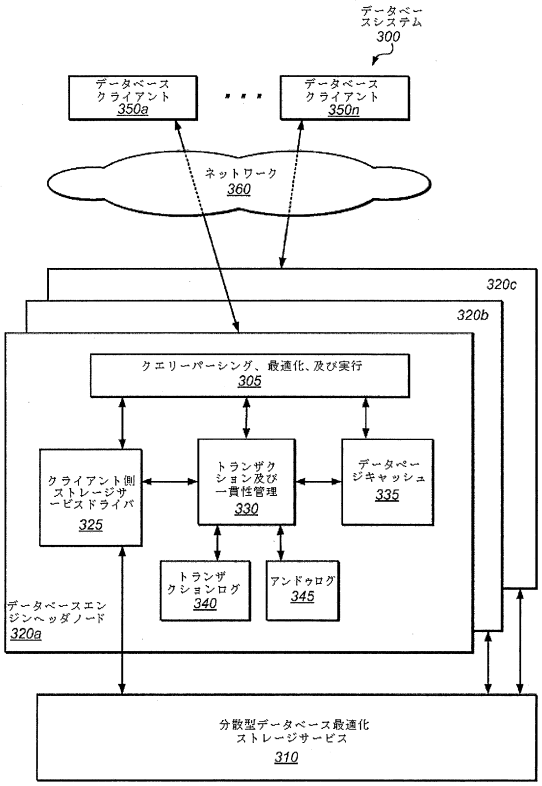
【図１】



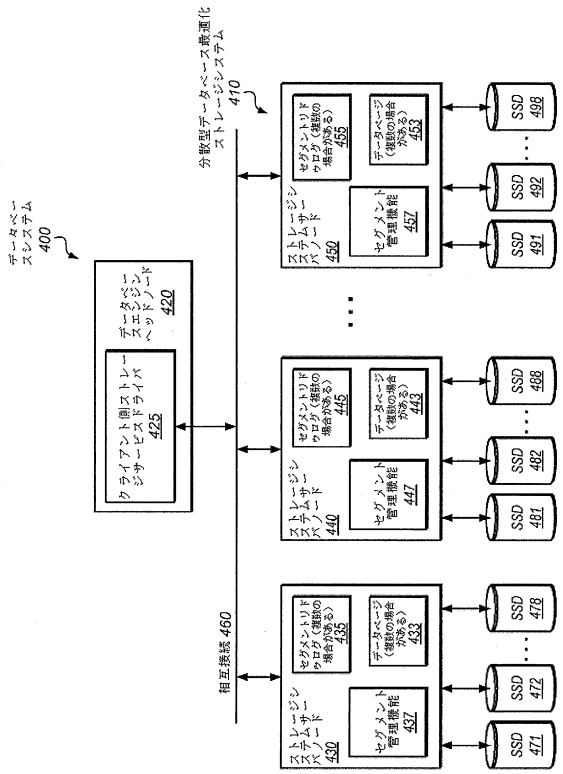
【図２】



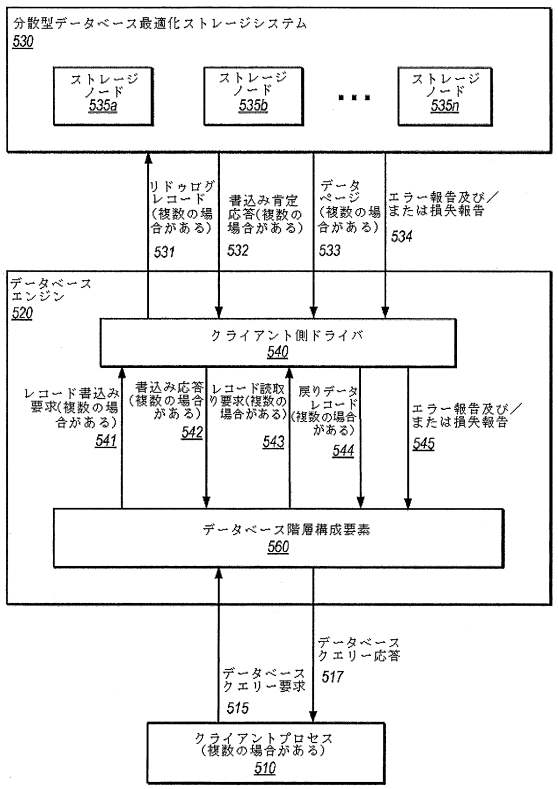
【図３】


データベース
システム
300
データベース
クライアント
350a
データベース
クライアント
350n
ネットワーク
360
320c
320b
クエリーパーシング、最適化、及び実行
305
クライアント側
ストレージサー
ビスドライバ
325
トランザク
ション及び
一貫性管理
330
データページ
キャッシュ
335
トランザ
クションログ
340
アンドゥログ
345
データベースエン
ジンヘッダノード
320a
分散型データベース最適化
ストレージサービス
310

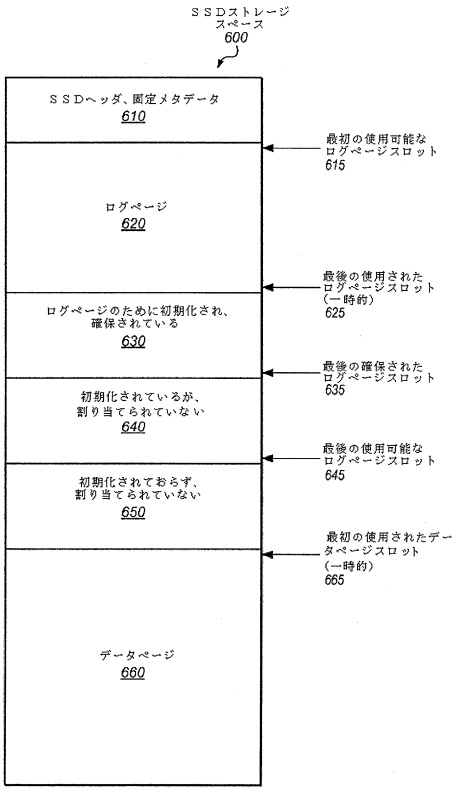
【図４】


分散型データベース最適化
ストレージシステム
410
データベース
システム
400
データエンジン
ヘッドノード
420
クライアント側ストレー
ジサービスドライバ
425
相互接続 460
ストレージシステムノード
450
ストレージシステムノード
440
ストレージシステムノード
430
セグメントリドゥログ（複数の場合がある）
455
データページ（複数の場合がある）
453
セグメント管理機能
457
セグメントリドゥログ（複数の場合がある）
445
データページ（複数の場合がある）
443
セグメント管理機能
447
セグメントリドゥログ（複数の場合がある）
435
データページ（複数の場合がある）
433
セグメント管理機能
437
SSD 498
SSD 492
SSD 491
SSD 488
SSD 482
SSD 481
SSD 478
SSD 472
SSD 471

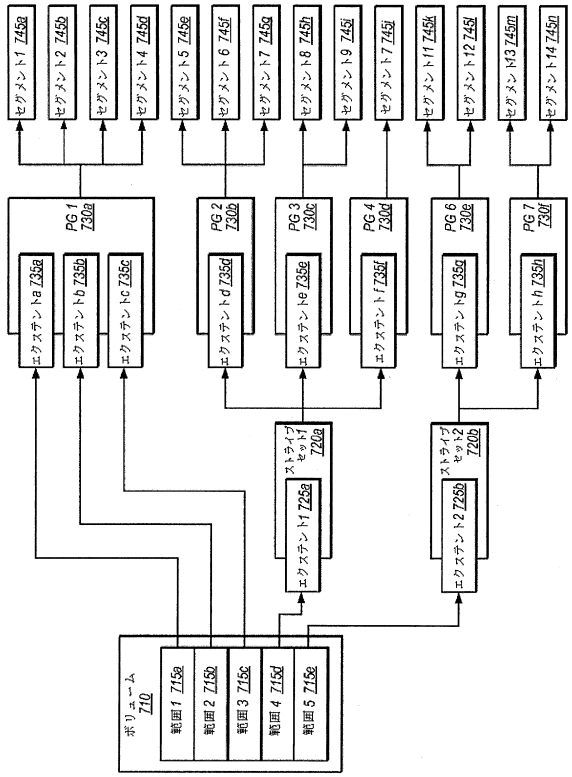
【図５】


分散型データベース最適化ストレージシステム
530
ストレージ
ノード
535a
ストレージ
ノード
535b
ストレージ
ノード
535n
リドゥログ
レコード
（複数の場
合がある）
531
書込み肯定
応答（複数の
場合がある）
532
データ
ページ
（複数の場
合がある）
533
エラー報告及び／
または損失報告
534
データベース
エンジン
520
クライアント側ドライバ
540
レコード書込み
要求（複数の場
合がある）
541
書込み応答
（複数の場合
がある）
542
レコード読取
り要求（複数の
場合がある）
543
戻りデータ
レコード
（複数の場合
がある）
544
エラー報告及び／
または損失報告
545
データベース階層構成要素
560
データベース
クエリー要求
515
データベース
クエリー応答
517
クライアントプロセス
（複数の場合がある）
510

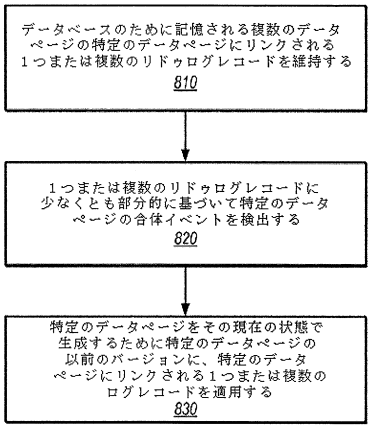
【図６】


ＳＳＤストレージ
スペース
600
ＳＳＤヘッダ、固定メタデータ
610
最初の使用可能な
ログページスロット
615
ログページ
620
最後の使用された
ログページスロット
（一時的）
625
ログページのために初期化され、
確保されている
630
最後の確保された
ログページスロット
635
初期化されているが、
割り当てられていない
640
最後の使用可能な
ログページスロット
645
初期化されておらず、
割り当てられていない
650
最初の使用されたデー
タページスロット
（一時的）
665
データページ
660

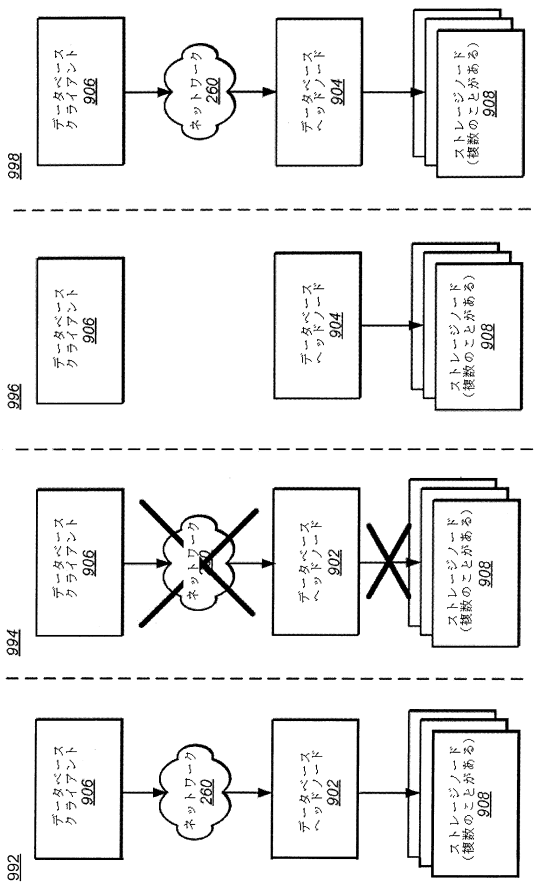
【図７】


セグメント1 745a
セグメント2 745b
セグメント3 745c
セグメント4 745d
セグメント5 745e
セグメント6 745f
セグメント7 745g
セグメント8 745h
セグメント9 745i
セグメント10 745j
セグメント11 745k
セグメント12 745l
セグメント13 745m
セグメント14 745n
PG 1 730a
PG 2 730b
PG 3 730c
PG 4 730d
PG 6 730e
PG 7 730f
エクステントa 735a
エクステントb 735b
エクステントc 735c
エクステントd 735d
エクステントe 735e
エクステントf 735f
エクステントg 735g
エクステントh 735h
ストライプセット1 720a
ストライプセット2 720b
エクステント1 725a
エクステント2 725b
ボリューム 710
範囲1 715a
範囲2 715b
範囲3 715c
範囲4 715d
範囲5 715e

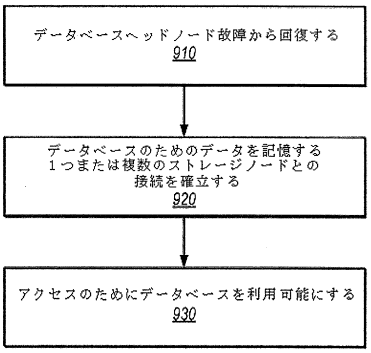
【図８】


データベースのために記憶される複数のデータページの特定のデータページにリンクされる1つまたは複数のリドゥログレコードを維持する
810
1つまたは複数のリドゥログレコードに少なくとも部分的に基づいて特定のデータページの合体イベントを検出する
820
特定のデータページをその現在の状態で生成するために特定のデータページの以前のバージョンに、特定のデータページにリンクされる1つまたは複数のログレコードを適用する
830

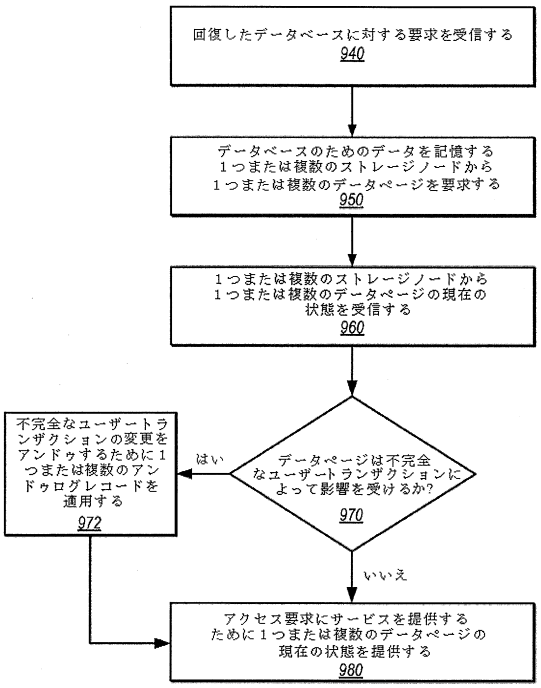
【図９Ａ】


998
996
994
992
データベースクライアント 906
ネットワーク 260
データベースヘッドノード 904
データベースヘッドノード 902
ストレージノード（複数のことがある） 908

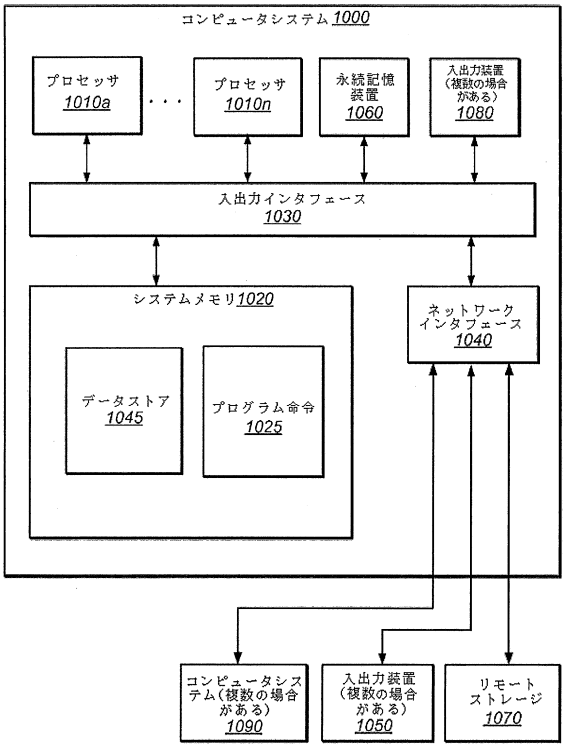
【図９Ｂ】


データベースヘッドノード故障から回復する
910
データベースのためのデータを記憶する1つまたは複数のストレージノードとの接続を確立する
920
アクセスのためにデータベースを利用可能にする
930

【図９Ｃ】


回復したデータベースに対する要求を受信する
940
データベースのためのデータを記憶する
1つまたは複数のストレージノードから
1つまたは複数のデータページを要求する
950
1つまたは複数のストレージノードから
1つまたは複数のデータページの現在の
状態を受信する
960
データページは不完全
なユーザートランザクションに
よって影響を受けるか?
970
はい
不完全なユーザートランザクションの変更を
アンドゥするために1
つまたは複数のアン
ドゥログレコードを
適用する
972
いいえ
アクセス要求にサービスを提供する
ために1つまたは複数のデータページの
現在の状態を提供する
980

【図１０】


コンピュータシステム 1000
プロセッサ
1010a
プロセッサ
1010n
永続記憶
装置
1060
入出力装置
(複数の場合
がある)
1080
入出力インタフェース
1030
システムメモリ1020
データストア
1045
プログラム命令
1025
ネットワーク
インタフェース
1040
コンピュータシス
テム(複数の場合
がある)
1090
入出力装置
(複数の場合
がある)
1050
リモート
ストレージ
1070

フロントページの続き

(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
(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　ファハン，ニール
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

審査官　漆原　孝治

(56)参考文献　特表２０１２－５０７０８６（ＪＰ，Ａ）
特開２００４－２２７１６９（ＪＰ，Ａ）
特開２００４－３４８７０１（ＪＰ，Ａ）
特開平０６－１９５２５０（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｇ０６Ｆ　１２／００