(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第5484470号
(P5484470)
(45)発行日　平成26年5月7日(2014.5.7)　(24)登録日　平成26年2月28日(2014.2.28)

(51)Int.Cl.　F I
G06F 17/30 (2006.01)　G06F 17/30 340D

請求項の数 18　(全 27 頁)

(21)出願番号	特願2011-528004 (P2011-528004)
(86)(22)出願日	平成21年9月18日 (2009.9.18)
(65)公表番号	特表2012-503256 (P2012-503256A)
(43)公表日	平成24年2月2日 (2012.2.2)
(86)国際出願番号	PCT/US2009/057526
(87)国際公開番号	WO2010/033834
(87)国際公開日	平成22年3月25日 (2010.3.25)
審査請求日	平成24年7月27日 (2012.7.27)
(31)優先権主張番号	61/192,668
(32)優先日	平成20年9月19日 (2008.9.19)
(33)優先権主張国	米国 (US)
(31)優先権主張番号	61/099,872
(32)優先日	平成20年9月24日 (2008.9.24)
(33)優先権主張国	米国 (US)

(73)特許権者　502303739
オラクル・インターナショナル・コーポレイション
アメリカ合衆国カリフォルニア州９４０６５レッドウッド・シティー，オラクル・パークウェイ５００
(74)代理人　110001195
特許業務法人深見特許事務所
(72)発明者　ポタポフ，ドミトリー
アメリカ合衆国、９４０６２　カリフォルニア州、エメラルド・ヒルズ、オーク・ノール・ドライブ、３５１５

最終頁に続く

(54)【発明の名称】オフロードされたブルームフィルタを伴うインテリジェントストレージにおける協調並列フィルタ処理を用いるハッシュジョイン

(57)【特許請求の範囲】
【請求項１】
データベースサーバが、データ記憶システムが第１のテーブルに関するデータを記憶する１つ以上のデータブロックを識別するステップを含み、
前記データ記憶システムは、前記第１のテーブルを含む論理構造に対するデータが、前記１つ以上のデータブロックを含む複数のブロック構造に記憶される、１つ以上の記憶装置を含み、さらに、
前記データベースサーバが、前記データ記憶システムに対して、
ａ）　前記データ記憶システムに対し、前記データベースサーバに前記１つ以上のデータブロックを与えるよう要求を送り、前記要求は前記１つ以上のデータブロックを具体的に識別し、さらに、
ｂ）　フィルタ処理条件を記述するメタデータを送るステップを含み、
前記要求は、前記データ記憶システムによって解釈されると、前記データ記憶システムに前記１つ以上のデータブロックを前記１つ以上の記憶装置から検索させる通信であり、
前記フィルタ処理条件を記述するメタデータは、前記データ記憶システムによって解釈されると、前記フィルタ処理条件を記述するメタデータを用いて前記検索された１つ以上のデータブロックの内容を少なくともフィルタ処理することにより、前記データ記憶システムに、フィルタ処理されたデータを発生させるデータであり、さらに、
前記要求に応答して、前記データベースサーバが前記フィルタ処理されたデータを前記データ記憶システムから受取るステップを含み、

前記フィルタ処理条件を記述するメタデータを前記データ記憶システムに送るステップの前に、第２のテーブルの１つ以上の属性に基づいてブルームフィルタを生成するステップとをさらに含み、

前記フィルタ処理条件を記述するメタデータを前記データ記憶システムに送るステップの前に、Ｉ／Ｏ帯域幅の利用可能性、前記データベースサーバおよび前記データ記憶システムにおける計算能力に基づいて、前記ブルームフィルタを前記データ記憶システムに送り出すべきかどうかを判断し、前記ブルームフィルタを送り出すべきと判断された場合に、前記ブルームフィルタに基づいて少なくとも１つの述語を生成するステップをさらに含み、前記フィルタ処理条件を記述するメタデータは前記少なくとも１つの述語を含む、方法。

【請求項２】

前記１つ以上のデータブロックを識別するステップは、前記１つ以上のデータブロックを構成するバイトからなる１つ以上の範囲が記憶される１つ以上の位置を識別するステップを含む、請求項１に記載の方法。

【請求項３】

前記フィルタ処理されたデータは１つ以上のデータブロックを含む、請求項２に記載の方法。

【請求項４】

前記データ記憶システムに前記第１のテーブルに関する論理構造の少なくとも一部を示すメタデータを送るステップをさらに含む、請求項１に記載の方法。

【請求項５】

前記フィルタ処理条件を記述するメタデータは、第２のテーブルの特性の内容を記述する、請求項１に記載の方法。

【請求項６】

前記データベースサーバが、前記フィルタ処理されたデータに基づいて、第１のジョイン演算を前記第１のテーブルおよび前記第２のテーブルにおいて実行するステップをさらに含む、請求項５に記載の方法。

【請求項７】

前記フィルタ処理条件を記述するメタデータは、さらに、第３のテーブルの１つ以上の特性を記述し、

前記フィルタ処理条件を記述するメタデータは、前記データ記憶システムによって解釈されると、前記メタデータに記述されるように、前記第３のテーブルの前記１つ以上の特性に基づいて前記１つ以上のデータブロックの内容をさらにフィルタ処理することにより、前記フィルタ処理されたデータを前記データ記憶システムに発生させるデータであり、

前記方法はさらに、前記データベースサーバが、前記第１のジョイン演算の結果に基づいて、前記第１のテーブル、前記第２のテーブルおよび前記第３のテーブルに対して第２のジョイン演算を実行するステップを含む、請求項６に記載の方法。

【請求項８】

前記フィルタ処理条件を記述するメタデータは１つ以上のジョインフィルタ処理条件を含み、

前記方法は、さらに、前記ジョイン演算を前記フィルタ処理されたデータに基づいて実行するステップを含む、請求項１に記載の方法。

【請求項９】

データ記憶システムが、

データベースサーバに１つ以上のデータブロックを提供するよう前記データ記憶システムに対する要求を受取るステップを実行し、前記要求は、前記１つ以上のデータブロックが前記データ記憶システムに記憶される１つ以上の位置を具体的に識別し、

前記データ記憶システムは、論理構造に対するデータが前記１つ以上のデータブロックを含む複数のブロック構造に記憶される１つ以上の記憶装置を含み、

前記論理構造は、データベーステーブルであり、前記ブロック構造とは異なる構造のも

のであり、さらに、
フィルタ処理条件を記述するメタデータを受取るステップと、
前記要求に応答して、前記１つ以上の記憶装置上の前記１つ以上の位置から前記１つ以上のデータブロックを読出すステップと、
前記フィルタ処理条件を記述するメタデータに基づいて、検索された１つ以上のデータブロックの内容を少なくともフィルタ処理することによって、フィルタ処理されたデータを発生させるステップと、
前記要求に応答するステップとを実行し、前記応答は前記フィルタ処理されたデータを含み、
前記フィルタ処理条件を記述するメタデータはブルームフィルタを含み、
前記データベースサーバにおいて、Ｉ／Ｏ帯域幅の利用可能性、前記データベースサーバおよび前記データ記憶システムにおける計算能力に基づいて、前記ブルームフィルタを前記データ記憶システムに送り出すべきかどうかを判断し、前記ブルームフィルタを送り出すべきと判断された場合には、前記ブルームフィルタに基づいて少なくとも１つの述語を生成するステップをさらに含み、前記フィルタ処理条件を記述するメタデータは前記少なくとも１つの述語を含む、方法。
【請求項１０】
前記データ記憶システムが、
データが前記１つ以上のデータブロックに記憶されるテーブルに関して論理構造の少なくとも一部を示すメタデータを受取るステップと、
前記論理構造の前記少なくとも一部に基づいて、前記１つ以上のデータブロックによって表現される１つ以上の属性値を識別するステップとを実行することをさらに含み、
前記１つ以上のデータブロックの内容をフィルタ処理するステップは、さらに、前記フィルタ処理条件を記述するメタデータにおける前記１つ以上のフィルタ処理条件を前記識別された１つ以上の属性値に適用するステップに基づく、請求項９に記載の方法。
【請求項１１】
前記応答は、さらに、前記１つ以上のデータブロックの少なくとも１つを含み、前記１つ以上のデータブロックの前記少なくとも１つはフィルタ処理されない、請求項９に記載の方法。
【請求項１２】
前記フィルタ処理されたデータを発生させるステップは、さらに、第１のデータブロックによって表現される１つ以上の論理行を、第２のデータブロックによって表現される１つ以上の論理行と結合することにより、前記データベースサーバに返されるべき１つ以上の結合された行を生成するステップを含む、請求項１０に記載の方法。
【請求項１３】
前記１つ以上のフィルタ処理条件は、前記要求されたデータに関して実行されるべきジョイン演算に対するフィルタ処理条件である、請求項９に記載の方法。
【請求項１４】
前記受取られたメタデータによって記述されるフィルタ処理条件は、前記１つ以上のデータブロックによって表現されるある特定の論理構造に関して実行されるべきある演算に対してである、請求項１～１３のいずれか１つに記載の方法。
【請求項１５】
命令を記憶する１つ以上の記憶媒体であって、前記命令は、１つ以上の計算装置によって実行されると、請求項１～１３のいずれか１つの方法の実行を引起す、記憶媒体。
【請求項１６】
データを複数のデータブロック構造に記憶する１つ以上のブロック化された記憶装置と、
データブロック構造に対する入力／出力（Ｉ／Ｏ）要求に応答して、前記１つ以上のブロック化された記憶装置に対してデータブロックの読出または書込を行なう記憶サーバと、

ネットワークを介して前記記憶サーバに接続され、第１のＳＱＬクエリをクライアントから受取り、前記第１のＳＱＬクエリに応答して、少なくとも、１）前記記憶サーバから、１つ以上のデータブロックからなる第１の集合を要求し、２）前記１つ以上のデータブロックからなる第１の集合を、少なくとも１つ以上のデータベーステーブルの１つ以上の列として解釈し、３）前記第１のＳＱＬクエリによって示される第１のデータベース動作を前記少なくとも１つ以上の列に対して実行することにより、結果集合を発生させるデータベースサーバとを含み、

第２のデータベース動作に関して、前記データベースサーバは、前記記憶サーバに対して、ａ）１つ以上のデータブロックからなる第２の集合を検索する１つ以上のアドレスを指定するＩ／Ｏ要求と、ｂ）フィルタ処理条件を記述するメタデータとを送るよう構成され、

前記記憶サーバは、前記Ｉ／Ｏ要求に応答して、前記１つ以上のデータブロックからなる第２の集合を、前記１つ以上の記憶装置から、指定される１つ以上のアドレスにおいて読出すよう構成され、

前記記憶サーバは、さらに、前記Ｉ／Ｏ要求に応答して前記１つ以上のデータブロックからなる第２の集合を読出すと、前記フィルタ処理条件を記述するメタデータに基づいて前記１つ以上のデータブロックからなる第２の集合の内容をフィルタ処理することにより、フィルタ処理されたデータを発生させるよう構成され、

前記記憶サーバは、さらに、前記Ｉ／Ｏ要求に応答して、前記フィルタ処理されたデータを前記データベースサーバに送るよう構成され、

前記データベースサーバは、さらに、前記記憶サーバにより返される前記フィルタ処理されたデータに基づいて前記第２のデータベース動作を実行するよう構成され、

前記第２のデータベース動作はジョイン演算であり、前記フィルタ処理条件を記述するメタデータは前記ジョイン演算に対する基準を含み、

前記ジョイン演算に対する前記基準はブルームフィルタを含み、

前記データベースサーバは、前記データベースサーバと前記記憶サーバの間のＩ／Ｏ帯域幅の利用可能性、前記データベースサーバおよび前記記憶サーバにおける計算能力に基づいて、前記ブルームフィルタを前記記憶サーバに送り出すべきかどうかを判断し、前記ブルームフィルタを送り出すべきと判断された場合には、前記ブルームフィルタに基づいて少なくとも１つの述語を生成し、前記フィルタ処理条件を記述するメタデータは前記少なくとも１つの述語を含む、システム。

【請求項１７】

前記記憶サーバは、さらに、前記データベースサーバから、前記１つ以上のデータブロックからなる第２の集合によって表現されるテーブルの１つ以上の特性を記述するメタデータを受取るよう構成され、

前記記憶サーバは、さらに、前記１つ以上のデータブロックからなる第２の集合を、前記テーブルの前記１つ以上の特性を記述するメタデータに基づいて第２の構造として解釈し、次いで、前記第２の構造を、前記フィルタ処理条件を記述するメタデータに基づいてフィルタ処理することによって、前記１つ以上のデータブロックからなる第２の集合の内容をフィルタ処理するステップを達成するよう構成される、請求項１６に記載のシステム。

【請求項１８】

前記記憶サーバおよび前記１つ以上の記憶装置は第１の計算装置にあり、前記データベースサーバは前記第１の計算装置にネットワーク接続される異なる計算装置にある、請求項１６に記載のシステム。

【発明の詳細な説明】

【技術分野】

【０００１】

発明の分野

この発明はデータ操作に関し、より特定的には、ジョイン演算におけるストレージ側の

参加に関する。
【背景技術】
【０００２】
このセクションに記載されるアプローチは、追求されるかもしれないアプローチではあるが、以前に構想または追求されたことがあるアプローチでは必ずしもない。したがって、そうではないと示されるのでなければ、このセクションに記載されるアプローチのいずれも単にそれらのこのセクションにおける包含により先行技術として適格である、と仮定されるべきではない。
【発明の概要】
【発明が解決しようとする課題】
【０００３】
データベースサーバ
一般に、データベースサーバなどのサーバは、統合されたソフトウェアコンポーネントと、計算資源、たとえばメモリやノードなどの割当との組合せであり、該ノードにおいて、統合されたソフトウェアコンポーネントを実行することに対して処理を行ない、そこにおいては、ソフトウェアと計算資源との組合せは、サーバのクライアントに代わってある特定のタイプの機能を提供することに対して専ら用いられる。データベースサーバは、１つ以上のデータベースに対するアクセスを支配し、それを容易にして、クライアントによる１つ以上のデータベースへのアクセスに対する要求を処理する。
【０００４】
データベースはデータおよびメタデータを含む。ハイレベルの観点からは、そのデータおよびメタデータは、論理構造内において、たとえば、リレーショナルおよび／またはオブジェクトリレーショナルデータベース構成に従って維持される。データベースメタデータは、テーブル、オブジェクトテーブル、ビュー、またはコンプレックスタイプなどのデータベースオブジェクトを定義する。ＳＱＬデータ定義言語（「ＤＤＬ」）命令をデータベースサーバに対して出すことにより、データベースオブジェクトを作成または構成する。
【０００５】
一般に、データは、データベース内において、１つ以上のデータコンテナとして論理的に構造化される。各コンテナはレコードを含む。各レコード内のデータは１つ以上のフィールドに組織化される。リレーショナルデータベースシステムでは、データコンテナは典型的にはテーブルと称され、レコードは行と称され、フィールドは列と称される。オブジェクト指向型データベースでは、データコンテナは典型的にはオブジェクトタイプまたはクラスと称され、レコードはオブジェクトと称され、フィールドは属性と称される。他のデータベースアーキテクチャは他の用語を用いるかもしれない。この発明を実現するシステムはどのような特定のタイプのデータコンテナまたはデータベースアーキテクチャにも限定されはしない。しかしながら、説明のため、ここに用いられる例および用語は、典型的には、リレーショナルまたはオブジェクトリレーショナルデータベースと関連付けられることにする。したがって、「テーブル」、「行」、および「列」という文言は、ここにおいては、それぞれ、データコンテナ、レコードおよびフィールドを指すよう用いられることにする。
【０００６】
記憶システム
データベースサーバは、データベースのための基底データを、１つ以上の永続的な記憶システムに維持する。これらの記憶システムは、典型的には、データベースサーバに対して大容量の永続的な不揮発性記憶を与え、そこにおいては、データベースサーバは、基底データを、しばしば、ハードディスクのような１つ以上の記憶装置の形態において記憶してもよい。そのような記憶システムの一例は記憶アレイである。
【０００７】
多くの記憶システムは、記憶装置バックアップ、記憶装置最適化、複数の個々の記憶装

置にわたるストライピング、共有されるデータアクセス、ブロックキャッシングなどの、ローレベルの専門化されたデータ管理機能を実行するためのソフトウェアおよびハードウェア論理で最適化される。したがって、データベースサーバは、しばしば、記憶システムに依存してそのようなローレベルの機能を提供し、データベースサーバはそれらの資源を他のタスク、たとえばクエリ編集および実行、データ解析、ならびにクライアントとの通信などに対して利用することができる。

【０００８】

数多くの実施例においては、データベースサーバによって利用される記憶システムは、１つそこそこ以上の単純な、線形アドレス指定される、ブロック化された、永続的な記憶装置を構成する。したがって、記憶システムは、それらが記憶する基底データによって表現される論理構造に気付かないかもしれない。さらに、データベースサーバと記憶システムとの間の相互作用は、バイトの範囲でディスクに対して読出または書込を行なう単純な入力／出力（Ｉ／Ｏ）要求に限られるかもしれない。

【０００９】

したがって、データベースサーバはデータベースデータをクライアントに対して上記のような論理構造として提示する一方で、データベースに対する基底データは、その記憶システムにおいて、異なる、より単純な構造で記憶されるかもしれない。たとえば、データベースサーバがデータを従来のハードディスクに記憶するためには、データベースサーバは、そのデータを、ハードディスクによりサポートされるブロック構造に一致するよう構造化する。したがって、たいていのクライアントは、データベースサーバに対し、テーブルおよび列などの論理構造に対する言及を介してデータを指し示す命令またはクエリを発行する一方、データベースサーバは実際にはそのデータを記憶システムから生ブロックで検索する。さまざまな記憶されるメタデータ、インデックスおよびヘッダの使用を介して、データベースサーバは、記憶システムから検索される構造内のデータを、論理テーブル、行および列として解釈することができる。利便性のため、データベースの「生」または基底データが記憶システムに記憶される構造は、以下、データブロックまたはデータ単位と称することにする。データブロックまたはデータ単位に関して記載される技術は、生データを記憶システムに記憶するための他の構造にも等しく適用可能であるとして理解されるべきである。

【００１０】

たとえば、あるデータベースサーバがあるデータベースにおけるある論理構造に対するアクセスを必要とするコマンドを実行する際、データベースサーバは、データをマッピングすることを利用して、ある記憶システムにおいて、その論理構造に対する基底データが記憶されるあるデータブロックまたはデータブロックからなるある範囲を識別してもよい。データベースサーバは、次いで、記憶システムに対し、マッピングされたデータブロックに対する読出要求を送ってもよい。この要求に応答して、記憶システムはその識別されたデータブロックをストレージから読出して、そ（れら）のデータブロックをデータベースサーバに送ってもよい。データベースサーバは、次いで、データブロックを、テーブルの論理行または列として解釈してもよい。データベースサーバは、データブロックを論理行および列とした自身の解釈に基づいて、コマンドを実行する。

【００１１】

データブロックを記憶システムに要求し検索するのに要する時間は、データベースサーバがデータベースコマンドを実行するのに必要とされる時間のうちの大きな量を表す。残念なことに、多くの演算においては、記憶システムから検索されるあるデータブロックのいくつかまたはすべてでさえもが、データベースコマンドの実行に関係がないかもしれない。たとえば、あるクライアントが、あるテーブルのある特定の列のみに対してデータを要求するかもしれない。テーブルが記憶システムに記憶される基底データブロック構造のため、データベースサーバは、単に、要求された列に対してだけではなく、他の列に対しても同様に、データからなるデータブロックを要求することを必要とされるかもしれない。データベースサーバは、次いで、非要求列に対するデータを破棄することになる。かく

して、非要求列に対するデータの転送は不要となった。

【００１２】

別の例として、あるクエリは、データベースサーバにデータを結果集合からフィルタ処理させるある述語条件を含むかもしれない。たとえば、そのクエリは、「色」列値が「赤」であるすべての行を要求するかもしれない。残念なことに、テーブルの基底データのすべてを検索することなしには、データベースは、どの行がそのような述語を満足させるか知るよしはない。たとえば、１００，０００行テーブルに、該述語を満足させるのはたった１つの行しかないかもしれない。その行を見つけ出すためには、データベースサーバは、それでもなお、記憶システムから、テーブルに対するすべてのデータブロックを、それらのほとんどがクエリとは無関係であることになっても、検索しなければならない。

【００１３】

ジョイン演算

データベースサーバは、通常、「ジョイン演算」として公知である、あるクラスの演算の実行を必要とするコマンドを受取る。一般的に言って、ジョイン演算は、第１のテーブルにおける各行を、あるジョイン基準に一致する第２のテーブルにおける各行とマージすることを含む。ジョイン演算に関与する各テーブルごとに、ジョイン演算は、ジョイン列またはジョイン属性として知られる１つ以上の列の部分集合を指定する。ジョイン基準は、第１のテーブルにおけるジョイン列と第２のテーブルにおけるジョイン列との比較を含んでもよい。ジョイン基準は、さらに、１つ以上の条件を含んでもよい。これらの条件は、しばしば、述語として表現され、したがって、ジョイン基準はジョイン述語と称されてもよい。

【００１４】

「等結合」演算は一般的なタイプのジョイン演算であり、そこにおいては、第１のテーブルの行Ａと第２のテーブルの行Ｂとのマージは、行Ａにおけるジョイン列の値が行Ｂのジョイン列の値と等価である場合にのみ行なわれる。たとえば、以下のＳＱＬ文は、「employee（従業員）」と名付けられたテーブルにおける各行と「Department（部署）」と名付けられたテーブルにおける各行とのマージが、双方の行がそれらのそれぞれの「Department ID」列に対して同じ値を共有する場合にのみ行なわれる、等結合演算を指定する。

【００１５】

【数１】

```
SELECT *
FROM  employee
    INNER JOIN department
      ON employee.DepartmentID = department.DepartmentID
```

【００１６】

「アンチ結合」は「等結合」と類似するあるタイプのジョイン演算であるが、そこにおいては、行のマージはそれらのそれぞれのジョイン列の値が等価でない場合においてのみ行なわれる。

【００１７】

データベースサーバは、通常は、「ハッシュジョイン」として公知のアルゴリズムに依存して、「等結合」演算および「アンチ結合」演算の両方を効率よく行なう。「ハッシュジョイン」演算は、一般的に言って、２つの段階を含む。「ビルド段階」として知られる第１の段階においては、データベースサーバは、ジョイン列においてハッシュ関数に従って第１のテーブルの各行をハッシングすることによりハッシュテーブルを生成する。「プローブ段階」として知られる第２の段階においては、データベースサーバは、次いで、第２のテーブルのキャッシュ行を反復的に走査する。第２のテーブルの各行ごとに、データベースサーバはハッシュ関数およびハッシュテーブルを用いて、第１のテーブルにおける等価なジョイン列値の行を識別する。一致する行が識別されると、それらの行をマージし

て、ジョイン演算に対する結果集合に加え、それらの行も任意の適用可能なジョイン述語と一致すると仮定する。

【００１８】

ジョイン演算における双方のテーブルが相対的に大きい場合、データベースサーバは、第１のテーブル全体に対するハッシュテーブルをメモリに当てはめることができないかもしれない。そのような状況においては、テーブルを区切り、データベースサーバは、ネスト化されたループを、それらの区切りに関して実行してもよい。しかしながら、２段階化されたアプローチは、ループのキャッシュ反復中において本質的に同じままである。

【００１９】

ブルームフィルタ

ブルームフィルタは、集合の要素ではない要素を識別するよう用いられる確率的データ構造である。ブルームフィルタは、ある度合いの誤りを考慮に入れることにより、考えられ得るタイプ値からなる非常に大きな集合の存在の状態がはるかにより小さなメモリで表現されることを可能にする。ブルームフィルタの性質は、ある要素に対するブルームフィルタの適用によって、（１）その要素は、ブルームフィルタが導出された集合にない、または（２）その要素は、ブルームフィルタが導出された集合にありそうな候補である、という２つの可能性のうちの１つが明らかにされるというものである。換言すると、ブルームフィルタの適用は、偽陽性をもたらす結果となるかもしれないが、偽陰性をもたらす結果とはけっしてならない。ブルームフィルタを生成し利用するための技術は周知であり、たとえば、"Space/time trade-offs in hash coding with allowable errors" by Burton H. Bloom, Communications of the ACM, Volume 13, Issue 7, 1970に記載される。

【００２０】

ブルームフィルタのある例示的実現例が続く。そのブルームフィルタは、しばしば、ビットの配列として実現される。配列における各項目は０に初期設定される。配列は、次いで、１つ以上のハッシュ関数を集合の各要素に適用することにより埋められ、各ハッシュ関数の結果は、ある配列項目のインデックスを識別して１にセットする。一旦ブルームフィルタがある集合に対して構築されると、次いで、ある要素を、そのブルームフィルタに対してテストする。要素をブルームフィルタに対してテストする処理は、たとえば、同じ１つ以上のハッシュ関数をその要素に適用し、その１つ以上のハッシュ関数をその要素に適用することにより識別されたインデックスのうちの任意のものにおける配列の値が依然として０にセットされるかどうかを判断することを含んでもよい。そうである場合には、その要素は、ブルームフィルタによって表現される集合にはないと判断される。

【００２１】

関与するデータによって、あるデータベースサーバは、「ブルームフィルタ」を、ジョイン演算を実行することに対して利用する。データベースサーバは、ジョイン演算の第１のテーブルに対して生成されるハッシュテーブルに基づいてブルームフィルタを生成してもよい。データベースサーバは、次いで、ブルームフィルタを利用して、ジョイン列の値が第１のテーブルに対するハッシュテーブルにおける空のフィールドにハッシングされる、第２のテーブルにおける１つ以上の行を識別してもよい。そのような行は、したがって、実際にハッシュテーブルを調べる必要なく、ジョイン演算における考慮から外してもよい。ハッシュテーブルルックアップは多くの場合において相対的に高価な演算であるので、この技術は、しばしば、ジョイン演算に対してより大きな効率性をもたらす結果となる。一方、ブルームフィルタを通過する行は、次いで、実際のハッシュテーブルと比較され、もしあれば、第１のテーブルの一致する行を具体的に識別する。

【００２２】

任意の所与のジョイン列に対する考えられ得る値の数は大きい－多くの場合事実上無限である－かもしれないため、ジョイン演算に対するハッシュテーブルは非常に大きく成長し得る。ブルームフィルタは、どのようなサイズのものでもあり得るが、通常は、サイズがハッシュテーブルよりもはるかに小さい。しかしながら、ブルームフィルタが効果的であるためには、ブルームフィルタはやはりサイズがかなり大きくなければならない。

【００２３】
この発明を、同様の参照番号は同様の要素を示す添付の図面を参照して、限定的にではなく、例示的に説明する。
【図面の簡単な説明】
【００２４】
【図１】ここに記載される技術が実現されてもよいシステムのブロック図である。
【図２】データベースサーバがデータベース動作を記憶システムにシフトしてもよい方法を示すフローチャートである。
【図３】記憶システムがデータベースサーバによって要求されたデータを前フィルタ処理してもよい方法を示すフローチャートである。
【図４】データベースサーバがジョイン演算のある局面を記憶システムにシフトする方法を示すフローチャートである。
【図５】この発明の実施例が実現されてもよい計算装置のブロック図である。
【発明を実施するための形態】
【００２５】
詳細な説明
以下の記載においては、説明のため、多数の具体的な詳細を、この発明の十分な理解を与えるために述べる。しかしながら、この発明はこれらの具体的な詳細なしに実施されてもよいことは明らかである。他の例では、周知の構造および装置が、ブロック図において、この発明を不要に不明瞭にすることを回避するために、示される。
【００２６】
１．０．概要
従来であればデータベースサーバによって実行されるであろうジョイン演算のさまざまな局面を実行すべく記憶システムにおいて処理資源を利用することによりデータベースシステムの効率性を向上させるための技術がここに記載される。ある実施例に従うと、データベースサーバは、記憶システムに対して、データ単位からなるある範囲に対する要求をなす場合に、その要求とともに、データが要求されているジョイン演算のさまざまな局面を記述するジョインメタデータを含むよう構成される。記憶システムは、その要求されたデータをディスクから通常どおり読出す。しかしながら、要求されたデータを記憶システムに送り返す前に、記憶システムはその要求されたデータをジョインメタデータに基づいてフィルタ処理し、ジョイン演算に無関係であると保証されるある量のデータを除去する。記憶システムは、次いで、そのフィルタ処理されたデータをデータベースサーバに返す。この技術の恩恵の中で、特に、データベースシステムは、記憶システムとデータベースサーバとの間におけるある量のデータの不要な転送を回避する。
【００２７】
ある実施例に従うと、ジョインメタデータは、ジョイン演算において第１のテーブルに対して生成されるハッシュテーブルおよび／またはそのようなハッシュテーブルに対するブルームフィルタを示してもよい。ある実施例に従うと、データベースサーバは、ブルームフィルタの述語への変換を、それを記憶システムに通信する前に行なう。データベースサーバは、このブルームフィルタ述語を、０またはそれより多い他のジョイン述語とともに、記憶システムに送る。記憶システムは、検索された範囲のデータ単位を、これらのジョイン述語に基づいてフィルタ処理するよう構成される。
【００２８】
ある実施例に従うと、記憶システムは、そこに記憶されるデータ単位おける生データについて、ある論理構造の少なくとも何かを識別するための論理とともに構成される。たとえば、記憶システムは、ヘッダ、データブロック行、およびデータブロック行フィールドなどのような一般的なデータブロック要素を認識するよう構成されてもよい。しかしながら、記憶システムは、要求されたデータがあるテーブルの名称、またはあるフィールドが属する列コンテナの名称などのような、要求されたデータ単位からなる完全な論理構造に必ずしも気付くというわけではない。データベースサーバからの要求は、したがって、デ

ータベースサーバがデータブロックのある局面をある論理構造にどのように変換するかを示すメタデータを含んでもよい。たとえば、データベースサーバは、ある要求された範囲のデータブロックにおける各データブロック行の第３のフィールドはジョイン列に対応する旨を示すメタデータを含んでもよい。

【００２９】

一般的なデータ単位構造を識別する論理、どのようにしてデータベースサーバはデータ単位のある局面をある論理構造に変換するかを示すメタデータ、およびハッシュテーブルまたはブルームフィルタを示すメタデータに基づいて、記憶システムは、ジョイン演算に無関係な行に対応するデータ単位の部分を識別してもよい。記憶システムは、これらの無関係な部分を、要求されたデータをデータベースサーバに送り返す前に、その要求されたデータから除去する。ある実施例では、記憶システムは、データ単位を、行集合、行ソース、またはテーブルなどのような論理構造に変換してもよい。変換された論理構造は、データベースサーバに対して、データ単位の代わりに、または仮想データ単位において、無関係の部分に対応するすべてのデータを差し引いて送られる。

【００３０】

２．０．構造的概要

図１は、ここに記載される技術が実施されてもよいシステムのブロック図である。図１のシステムおよび以下に記載される他の例示的実施例は、しかしながら、ここに記載される技術が実施されてもよい多数の異なるシステムのいくつかの例である。

【００３１】

図１を参照して、記憶システム１００は、いくつかの異なるクライアント１３０、１３２、１３４、１３６および１３８によって用いられるデータに対するストレージを提供する。それらのクライアントは、データベースコマンドをデータベースサーバ１２０、１２２および１２４に送ることにより、記憶システム１００を間接的に利用する。図１のシステムはわずかに１つの記憶システムを示しているが、他の実施例では、データベースサーバ１２０～１２４は、複数の記憶システムに依存して、Ｉ／Ｏ要求を記憶システムにわたって並列に分散して、クライアント１３０～１３８により要求されるデータを提供してもよい。

【００３２】

２．１．例示的データベースサーバおよびクライアント

データベース１２０～１２４は、たとえば、データベースコマンドに応答して、記憶システム１００において記憶装置１０２および１０４にわたって広がるデータベースに記憶されるデータを記憶、検索および操作するための動作を実行するデータベースサーバであってもよい。クライアント１３０～１３８は、たとえば、データベースコマンドをデータベースサーバ１２０～１２４に送るデータベースアプリケーションであってもよい。クライアント１３０～１３８およびデータベースサーバ１２０～１２４は１つ以上の計算装置によって実現されてもよい。クライアント１３０～１３８は、各々、ネットワークまたはローカルインターフェイスを介する通信を含むさまざまな手段のうちの任意のものを介して、データベースサーバ１２０～１２４の１つ以上と対話してもよい。例示される実施例では、クライアント１３０および１３２はデータベースサーバ１２０と対話し、クライアント１３４はデータベースサーバ１２２と対話し、クライアント１３６および１３８はデータベースサーバ１２４と対話する。実際の実現例では、データベースサーバと同時に対話するクライアントの数およびタイプはおそらく変動することになる。

【００３３】

上で示唆されるように、クライアント１３０～１３８とデータベース１２０～１２４との間における対話は、典型的には、データベースおよびテーブルのような論理構造の形式におけるデータの通信を伴う。クライアント１３０～１３８は、ＳＱＬ文などのようなデータベースコマンドをデータベースサーバ１２０～１２４に送り、各コマンドはこれらの論理構造のうちのあるものを参照する。これらのデータベース文の多くに応答するため、データベースサーバ１２０～１２４は、記憶システム１００から、参照される論理構造に

対応する生データを要求し、その生データをその参照される論理構造に変換し、データベース文によって示される任意の演算をその変換に基づいて行なわなければならない。多くの場合において、データベースサーバ１２０～１２４は、参照される論理構造に関して演算を行なった結果に基づく結果集合として知られる論理構造を、クライアント１３０～１３８に送り返す。

【００３４】

データベースサーバ１２０～１２４は、各々、さまざまな手段のうちの任意のものを介して、記憶システム１００と対話してもよい。たとえば、図１に示されるように、各データベースサーバ１２０～１２４はスイッチ１１０を介して記憶システム１００に接続される。スイッチ１１０は、たとえば、データベースサーバ１２０～１２４の各々に接続するための１つ以上のインターフェイスを伴うインフィニバンドスイッチであってもよい。スイッチ１１０は、さらに、記憶システム１００に接続するための１つ以上のインターフェイスを特徴としてもよい。他の実施例は、幅広いさまざまな通信リンクのうちの任意のものを介する接続、およびデータベースサーバ１２０～１２４におけるポートと記憶システム１００におけるポートとの間の接続を含む、記憶システム１００に対する接続のための他の手段を特徴としてもよい（したがってスイッチ１１０を必要とはしない）。

【００３５】

データベースサーバ１２０～１２４と記憶システム１００との間の通信は、Ｓｅｒｉａｌ　ＡＴＡまたはｉＳＣＳＩのような標準的なＩ／Ｏプロトコルを用いる単純な読出／書込要求を含む、さまざまな形式をとってもよい。たとえば、あるデータ単位においてあるアドレスにある生データがある演算を実行するのに必要であると判断したことに応答して、データベースサーバ１２０は記憶システム１００に対する単純な読出／書込要求を構築してもよい。応じて、記憶システム１００はその要求されたデータ単位で応答してもよい。

【００３６】

この発明のある実施例に従うと、データベースサーバ１２０～１２４は、さらに、記憶システム１００に対してエンハンストＩ／Ｏ要求を通信してもよい。これらの要求は、データベースサーバ１２０～１２４によって必要とされる生データを記憶するデータ単位の位置のみならず、生データをデータベースサーバ１２０～１２４に返す前にフィルタ処理するよう記憶システム１００によって実行されてもよい演算を記述するさまざまなメタデータを識別する。そのようなメタデータは、たとえば、ＳＱＬ述語、およびデータ単位における生データの論理構造のある局面を記述するメタデータを含んでもよい。データベースサーバ１２０～１２４は、次いで、エンハンストＩ／Ｏ要求に応答する記憶システム１００からフィルタ処理されたデータ単位を受取ってもよい。ある実施例では、データベースサーバ１２０～１２４によって受取られた、フィルタ処理されたデータ単位は、行フォーマット化されたデータなどのような、他の構造で再フォーマット化されることさえあってもよい。

【００３７】

ある実施例では、エンハンスト要求および応答は、拡張Ｉ／Ｏプロトコルを介して通信されてもよい。そのような通信に好適な例示的プロトコルはオラクルのｉＤＢプロトコルであり、それは、次いで、ゼロロスゼロコピーデータグラムプロトコル（ＺＤＰ）として公知の信頼性のあるデータグラムソケットプロトコルに基づいている。

【００３８】

２．２．例示的記憶システム

記憶システム１００は、データを記憶し、管理し、データに対するアクセスを与えるためのシステムである。ある実施例では、記憶システム１００は、データベースサーバ１２０～１２４が実現される１つ以上の計算装置とは物理的に別個である、自己充足的計算装置である。たとえば、記憶システム１００はＥｘａｄａｔａセルであってもよい。しかしながら、他の実施例では、記憶システム１００とデータベースサーバ１２０～１２４との間の物理的区別は必要ではない。

【００３９】
記憶システム１００は少なくとも記憶サーバコンポーネント１０６ならびに複数の記憶装置１０２および１０４を含む。記憶装置１０２および１０４は、データブロック、エクステント、およびセグメントなどのデータ単位に組織化される生データが記憶される、永続的な不揮発性メモリである。たとえば、各記憶装置１０２および１０４は従来のハードディスクであってもよい。
【００４０】
記憶サーバ１０６は、記憶装置１０２～１０４に記憶されるデータを記憶し、管理し、およびそれに対するアクセスを与えるための論理を実現する１つ以上のサブコンポーネントを含む。これらのサブコンポーネントは、たとえば、記憶システム１００において１つ以上のプロセッサにおいて実行されるハードウェアおよび／または１つ以上のソフトウェア処理によって実現されてもよい。記憶サーバ１０６は、たとえば、装置１０２～１０４に記憶されるデータに関して従来のデータＩ／Ｏ動作および管理タスクを実現するための記憶コントローラおよび／または記憶ドライバを含むさまざまなサブコンポーネントを含んでもよい。
【００４１】
別の例示的サブコンポーネントとして、記憶サーバ１０６は、データベースサーバ１２０～１２４および／またはスイッチ１１０に接続するための１つ以上のインターフェイスを含んでもよい。この１つ以上のインターフェイスは、データベースサーバ１２０～１２４に関して上で論じたように、要求と応答生データまたはフィルタ処理されたデータとの交換を容易にする。
【００４２】
記憶サーバ１０６は、さらに、１つ以上のインターフェイスに結合されるデータ処理サブコンポーネントを含んでもよい。このデータ処理サブコンポーネントは、受取られるＩ／Ｏ要求において識別されるように、生データを記憶装置１０４または１０６におけるある位置から読出すための論理を実現してもよい。Ｉ／Ｏ要求とともに受取られるメタデータに基づいて、データ処理サブコンポーネントは、さらに、生データを１つ以上のインターフェイスを介して返す前にフィルタ処理するための演算を実行してもよい。たとえば、サーバ１０６は、Ｉ／Ｏ要求とともに、要求された生データに当てはまるようＳＱＬ述語を示すメタデータを受取っていてもよい。サーバ１０６は、述語と一致しないすべての生データをフィルタ処理して除去してもよい。
【００４３】
ある実施例では、さまざまなフィルタ処理演算の実行は、データ処理サブコンポーネントによる生データの解釈が、データベースサーバ１２０～１２４によってその生データに割当てられる論理構造の少なくともある部分を認識するような態様で行われることを必要としてもよい。たとえば、データ処理サブコンポーネントは、データ単位のうちのどのバイトがデータブロックヘッダに対応するかを理解するための論理を必要としてもよい。データ処理ユニットは、さらに、データブロックヘッダに基づいて、データブロックのどのバイトがデータブロック行および／または論理行に対応するかを判断するための論理を必要としてもよい。データ処理ユニットは、さらに、各データブロック行または論理行においてフィールドを識別するための論理を必要としてもよい。データ処理サブコンポーネントは、次いで、識別されたそのデータの論理特性に対する自身の理解を利用して、たとえば、フィールドがあるフィルタ処理条件を満たさないデータブロック行に関してデータをフィルタ処理してもよい。
【００４４】
ある実施例では、データ処理コンポーネントは、さらに、Ｉ／Ｏ要求に関連付けられるメタデータを利用して、データ単位の論理構造に対する理解を助けてもよい。たとえば、行を、それらの「Ａｄｄｒｅｓｓ」列の値に基づいてフィルタ処理する述語を適用するために、データ処理コンポーネントは、各データ単位のどのフィールドが「Ａｄｄｒｅｓｓ」列に対応するのかを認識することができなければならない。この目的のため、データベ

ースサーバ１２０～１２４は、記憶システム１００に対し、どの番号付けされた列が「Ａｄｄｒｅｓｓ」とラベル付けされるかを示すメタデータを送っていてもよい。

【００４５】

データベースサーバ１２０～１２４によって生データに割当てられる論理構造の少なくとも一部を認識するための、およびある論理構造に基づいてフィルタ処理演算を実行するための論理は、データベースライブラリコンポーネントによって与えられてもよい。ある実施例に従うと、このデータベースライブラリコンポーネントは、データベースサーバ１２０～１２４によって実現される論理の部分集合を実現するための命令を含む。データベースライブラリコンポーネントは、記憶サーバ１０６が、データベースサーバによって通常実行される動作からなるある特別な部分集合を実行するのに必要である論理のみを含むよう最適化されてもよい。この特別な部分集合の動作は、たとえば、記憶サーバ１０６において効率よく実行され得る動作のみであってもよい。

【００４６】

たとえば、データベースライブラリコンポーネントは、データ単位を論理構造として解釈するための命令、それらの論理構造を述語に基づいてフィルタ処理するための命令、およびあるタイプのデータを集合させるための命令を含むかもしれないが、ある種のまたは他のデータベース動作を実行するための命令を欠くかもしれない。この動作の部分集合は実施例によって変動してもよく、たとえば、記憶システム１００は、データベースサーバ１２０～１２４に利用可能な資源よりもあるデータベース動作に対して適した資源を含んでもよく、したがって、データベースライブラリコンポーネントはそれらの動作のみに対する命令を含んでもよい。他の例として、データベースサーバ１２０～１２４が、Ｉ／Ｏ要求を、同じデータベースをミラーリングする複数の記憶システム１００にわたって分散させる実施例では、任意の所与の記憶システム１００が、あるテーブルに関する生データの一部にアクセスするのみであってもよい。したがって、各記憶システム１００は、テーブルのうちのわずかに１つの行を巻き込む動作を効率よく実行できるであろう一方で、記憶システム１００を用いて、そのテーブルのすべての行にアクセスすることを伴う動作を実行することは、それほど効率的ではないかもしれない。したがって、記憶サーバ１０６のデータベースライブラリコンポーネントにおける動作の部分集合は、テーブルの複数の行に対するアクセスを伴う動作に対する命令を省略してもよい。

【００４７】

上記の記憶システム１００のさまざまなコンポーネントおよびサブコンポーネントは、記載される技術を実施することができる記憶システムに対する構造上のアーキテクチャのほんの一例を示す。他の記憶システムでは、ここに記載される機能性は、異なる組のコンポーネントおよびサブコンポーネントによって提供されてもよい。それどころか、サブコンポーネント間のワークの区分が実施例ごとに異なってもよい。したがって、記憶システム１００の任意のコンポーネントまたはサブコンポーネントによって実行されるとして上に記載されるどのステップも、これ以降は、概して記憶システムまたは記憶サーバに帰することになる。

【００４８】

３．０．機能的概要

３．１．データベースサーバワークフロー

図２は、この発明のある実施例に従って、データベースサーバがデータベース動作を記憶システムにシフトしてもよい方法を示すフローチャート２００である。図２は、しかしながら、ここに記載される技術の１つの例示的実現例である。他の実施例ではより少ないかまたは追加のステップを特徴としてもよく、あるステップは異なる順序で実行されてもよい。

【００４９】

ステップ２１０において、データベースサーバはクライアントからデータベースコマンドを受取る。たとえば、データベースサーバ１２０はクライアント１３０からＳＱＬ文を受取る。

【００５０】
データベースコマンドに応答して、ステップ２２０において、データベースサーバは、データベースコマンドが含意する論理構造を識別してもよい。たとえば、データベースサーバ１２０は、コマンドの実行がテーブルＴ１からのデータを必要とすると判断してもよい。
【００５１】
含意される論理構造を識別したことに応答して、ステップ２３０において、データベースサーバは、論理構造に対する生データがある、記憶システムにおける１つ以上のデータ単位の１つ以上のアドレスを判断する。判断されるアドレスは、記憶システムが、それらのアドレスに対応する、記憶システムにおける物理的位置を識別することができる限りにおいて、論理的であってもよくまたは物理的であってもよい。たとえば、データベースサーバ１２０は、テーブルＴ１に対するデータブロックがある、記憶システム１００におけるあるエクステントの論理アドレスを示すデータをマッピングすることを維持してもよい。ある実施例では、データベースサーバ１２０は、さらに、この論理アドレスを、記憶装置１０４または１０６のうちの具体的な１つにおいてある物理アドレスに変換してもよい。
【００５２】
データベースコマンドにさらに応答して、ステップ２３５において、データベースサーバは、論理構造に当てはまるよう１つ以上のフィルタ処理条件を識別してもよい。たとえば、クライアント１３０からのデータベースコマンドは、クライアントに戻るようＴ１からある行の部分集合のみを判断するようテーブルＴ１の各行に対して評価されるべき述語を含んでいてもよい。別の例として、データベースコマンドはジョイン演算を指定していもよい。データベースサーバは、どのＴ１がフィルタ処理されてもよいかに基づいて、ジョイン演算のある特性を記述するジョインメタデータを生成してもよい。ジョインメタデータを生成するための例示的技術はセクション３．３および４．３で論ずる。
【００５３】
ステップ２４０において、データベースサーバはＩ／Ｏ要求を記憶システムに送る。Ｉ／Ｏ要求は、１つ以上のデータ単位を、ステップ２３０にて判断される１つ以上のアドレスによって識別する。たとえば、データサーバ１２０はｉＤＢプロトコル要求を記憶システム１００に対してスイッチ１１０を介して送り、テーブルＴ１に対応するエクステントを読出してもよい。
【００５４】
ステップ２５０において、データベースサーバは、記憶システムに対し、ステップ２３５にて識別された１つ以上のフィルタ処理条件を記述するメタデータを送る。このメタデータはステップ２４０の要求において送られてもよい。このメタデータは、さらに、ステップ２４０のＩ／Ｏ要求の前または後に送られてもよい。そのような場合、メタデータは、記憶システムにとってアクセス可能な位置に記憶され、その後、Ｉ／Ｏ要求に含まれるルックアップ識別子を用いて記憶システムにより検索されてもよい。
【００５５】
ステップ２６０において、Ｉ／Ｏ要求に応答して、データベースサーバはフィルタ処理されたデータを記憶システムから受取る。たとえば、データベースサーバから送られたＩ／Ｏ要求およびメタデータの自身の解釈のため、記憶システムは、１）１つ以上のデータ単位を記憶システムの永続的なストレージにおける指定されたアドレスから検索するステップ（このステップは、ある場合においては、データ単位を記憶システム内のキャッシュから検索することによって実行されてもよい）、および２）１つ以上のフィルタ処理条件を、検索された１つ以上のデータ単位に適用するステップ、のようなステップを実行していてもよい。この場合、データベースサーバによって受取られたフィルタ処理されたデータは、１つ以上のフィルタ処理条件を適用することに基づいてデータを１つ以上のデータ単位から除去する結果であろう。ステップ２６０において受取られるフィルタ処理されたデータが生成されていてもよい例示的ステップは、セクション３．２および３．３にて論

じられる。
【００５６】
ある実施例に従うと、フィルタ処理されたデータは元の１つ以上のデータ単位（たとえばデータブロック）と同じ形式であるが、あるデータが除去されている（たとえばフィルタ処理条件に一致し損ねたデータブロック行）。別の実施例では、データベースサーバは、１つ以上のデータ単位の元の構造以外の構造で返されるフィルタ処理されたデータを認識するよう構成されてもよい。たとえば、フィルタ処理されたデータは、ｉＤＢプロトコルを介して、ある論理構造において－行ソース、行集合またはテーブルなど－、フィルタ処理条件に一致しなかったあるデータなしで、返されてもよい。ある実施例では、フィルタ処理された論理構造は「仮想データブロック」内にラッピングされてもよい。仮想データブロックは、たとえば、他のデータブロックと同様の構造－あるヘッダおよび／または送信情報など－を含んでもよいが、ただし、ある行集合をそのペイロードとして伴う。
【００５７】
ステップ２７０において、もし必要であれば、データベースサーバはそのフィルタ処理されたデータにおいて追加の演算を実行する。たとえば、データベースサーバは、どのような理由であれ、データベースサーバが記憶システムに送り出さないと決めたフィルタ処理されたデータに対してあるデータベース動作を実行してもよい。別の例では、記憶システムは、フィルタ処理されたデータとともに、記憶システムはあるフィルタ処理条件をわずかに部分的にのみ適用したかまたは全く適用しなかった旨およびデータベースサーバはしたがってそれらのフィルタ処理条件をそれ自身で適用するべきである旨を示す情報を通信していてもよい。別の例では、データベースサーバは、記憶システムからＴ１に関するデータを要求し、そのデータを、データベースサーバが異なるＩ／Ｏ要求においてデータを要求した別のテーブルＴ２と結合できていてもよい。データベースサーバは、したがって、Ｔ１の無関係な行の（もしすべてではないとしても）ほとんどがフィルタ処理されたデータから除去されたという知識を持って、ジョイン演算の最後のステップを、フィルタ処理されたデータに基づいて実行してもよい。
【００５８】
ステップ２８０において、ステップ２６０および／またはステップ２７０のフィルタ処理されたデータに基づいて、データベースサーバは、データベースコマンドに対する結果集合を生成し、それをクライアントに返してもよい。
【００５９】
３．２．記憶システムワークフロー
図３は、この発明の一実施例に従う、記憶システムがデータベースサーバによって要求されたデータを前フィルタ処理してもよい方法を示すフローチャート３００である。図３は、しかしながら、ここに記載される技術の１つの例示的実現例である。他の実施例ではより少ないかまたは追加のステップを特徴としてもよく、あるステップは異なる順序で実行されてもよい。
【００６０】
ステップ３１０において、記憶システムにおけるある記憶サーバはデータを検索するようＩ／Ｏ要求を受取る。この要求は、要求されるデータが記憶システムに記憶される１つ以上のデータ単位の１つ以上の位置を識別する。たとえば、記憶サーバ１０６により受取られる要求は、データブロックからなるある範囲に対応する記憶装置１０２上の物理アドレスからなるある範囲を識別してもよい。そのような要求は、たとえば、図２のステップ２４０を実現するデータベースサーバ１２０によって送られていてもよい。または、その要求は、記憶サーバが物理アドレスに変換することができる論理アドレスを指定してもよい。
【００６１】
ステップ３２０において、記憶サーバは、ステップ３１０において要求されたデータの論理構造に関して実行されるべき演算に対する１つ以上のフィルタ処理条件を記述するメタデータを受取る。たとえば、メタデータは、要求されたデータによって表現される論理

列の値において条件付けられる１つ以上のＳＱＬ述語を含んでもよい。別の例として、メタデータは、要求されたデータによって表現されるテーブルで実行されるジョイン演算のある局面を記述してもよい。そのような要求は、たとえば、図２のステップ２５０を実現するデータベースサーバ１２０によって送られていてもよい。別の実施例では、このメタデータは、ステップ３１０のＩ／Ｏ要求において受取られてもよい。ある実施例では、このメタデータは、データベースサーバまたは他のソースから別途受取られ、記憶システムにとってアクセス可能な位置に記憶されていてもよい。そのような実施例では、Ｉ／Ｏ要求は、記憶システムが適切なメタデータを見つけ出してもよい識別子を含んでもよい。

【００６２】

ある実施例では、メタデータは、データベースサーバによって、またはデータベースサーバからの移動中に、直列化またはエンコードされていてもよい。そのような実施例では、ステップ３２０は、したがって、さらに、受取られたメタデータを再構築および／またはデコードすることを必要とする。

【００６３】

ステップ３３０において、Ｉ／Ｏ要求に応答して、記憶サーバは、１つ以上のデータ単位を、示された１つ以上の位置から読出す。たとえば、記憶サーバ１０６は、読出動作を記憶装置１０２において実行して、データブロックからなる範囲をその示された物理アドレスからフェッチしてもよい。

【００６４】

ステップ３４０において、ステップ３２０のメタデータはＩ／Ｏ要求に含まれたかまたは関連付けられたことを認識したことに応答して、記憶サーバは、ステップ３３０において読出された１つ以上のデータ単位および１つ以上のフィルタ処理条件に基づいて、フィルタ処理されたデータを生成する。ある実施例では、記憶サーバは、１つ以上のデータ単位のワーキングコピーからデータのうちの少なくとも何かを除去することによって、フィルタ処理されたデータを生成し、そのワーキングコピーは、次いで、そのフィルタ処理されたデータを構成する。別の実施例では、記憶サーバは、要求側データベースサーバが１つ以上のデータ単位をどのように解釈することになるかを少なくとも部分的に反映する論理構造に従うように、１つ以上のデータ単位を変換する。記憶サーバは、次いで、あるデータを論理構造から除去し、それによって、フィルタ処理されたデータを産み出す。

【００６５】

いずれの実施例においても、記憶サーバは、ステップ３２０のメタデータに記述される１つ以上のフィルタ処理条件に基づいて除去されるべきデータを識別する。フィルタ処理されたデータがどのように構築されるかに関わらず、除去されるべきデータを識別することは、ステップ３４０のサブステップ３４２において、記憶サーバがデータ単位におけるさまざまなデータを論理構造として解釈して、１つ以上のデータ単位のどの部分が１つ以上のデータ単位から前フィルタ処理されるべき論理構造に対応するかを識別できるようにすることを必要としてもよい。たとえば、記憶サーバ１０６は、データブロックのうちのある部分を、論理行に対応するとして識別し、それらの部分をそれに従って論理行に変換してもよい。

【００６６】

１つ以上のデータ単位を解釈するのを助けるために、データ単位を論理構造に変換するためのデータサーバの一般的な論理の部分集合が、記憶サーバにとって、たとえば、データベースライブラリコンポーネントにおける命令として利用可能にされてもよい。さらに、ステップ３２０において受取られたメタデータは、どのようにしてデータベースサーバが１つ以上のデータ単位－たとえばテーブル名称、列名称、フィールドタイプ、フィールドサイズなど－を解釈するかに関するさらなる情報を含んでもよい。

【００６７】

ステップ３４０のサブステップ３４４において、フィルタ処理条件を、ステップ３４２において識別された論理構造に適用する。フィルタ処理条件に一致しない論理構造は、ステップ３４６においては、Ｉ／Ｏ要求に対する結果からフィルタ処理して除去される。

【００６８】
ステップ３４２～３４６の一例として、記憶サーバ１０６は、命令をそのデータベースライブラリコンポーネントにおいて実行して、ステップ３４０の間に読出されたデータブロックにおいてヘッダ情報を見つけ出してもよい。このヘッダ情報に基づいて、記憶サーバ１０６は命令をデータベースライブラリにおいて実行して、テーブル行に対応するデータブロックの部分を見つけ出してもよい。記憶サーバ１０６はこれらの部分の各々をテーブル行に変換してもよい。記憶サーバ１０６に通信されるフィルタ処理条件のうちの１つは述語“Ｙｅａｒ＞２０００”であってもよい。記憶サーバ１０６は、したがって、ステップ３２０の間に送られたメタデータを利用して、各変換されたテーブル行のどの行が“Ｙｅａｒ”に対応するか判断してもよい。この知識に基づいて、記憶サーバ１０６は、変換された行のうちのどれが２０００より大きいＹｅａｒ列値を有さないか（つまり変換された行のどれが述語に一致しないか）を識別する。すべてのそのような行は、Ｉ／Ｏ要求から返される結果から前フィルタ処理される。

【００６９】
ステップ３５０において、記憶サーバは、Ｉ／Ｏ要求に対し、フィルタ処理されたデータとともに返答する。記憶サーバは、そのフィルタ処理されたデータを、記憶システムにおける永続的なストレージにおける生データの元の構造で構造化してもよく、または、行集合、行ソース、もしくはテーブルなどのような論理構造に従って構造化してもよい。さまざまな実施例においては、その返答は、さらに、さまざまな他のメタデータと並んで、記憶サーバによって実行されるフィルタ処理演算についての情報を含んでもよい。

【００７０】
ある実施例に従うと、ステップ３２０において受取られるメタデータは、さらに、データベースサーバによって必要とされる列の部分集合を示す。したがって、ステップ３４０において、記憶サーバは、さらに、それらの列に対応しないすべてのデータを除去してもよい。

【００７１】
３．３．ジョインフィルタ処理基準
ある実施例に従うと、データベースサーバは、ジョイン演算に対して必要とされるあるタスクを記憶システムにシフトしてもよい。たとえば、上においてステップ３２０および２５０において記憶システムに通信されるフィルタ処理条件は、ジョイン演算のある局面を記述する情報を含んでもよい。ある実施例では、記憶システムにシフトされるタスクは、他のジョインテーブルにおけるどの行にも一致しないと判断されるある行に対応するすべてのデータを取除くことによってジョイン演算に関与するテーブルのうちの１つに関してデータを前フィルタ処理することを含む。

【００７２】
図４は、この発明のある実施例に従う、データベースサーバがジョイン演算のある局面を記憶システムにシフトする方法を示すフローチャート４００である。図４は、しかしながら、ここに記載される技術の１つの例示的実現例である。他の実施例ではより少ないかまたは追加のステップを特徴としてもよく、あるステップは異なる順序で実行されてもよい。

【００７３】
ステップ４１０において、ジョイン演算の実行を開始したことに応答して、データベースサーバは、１つ以上の記憶システムと通信して、ジョイン演算に関与する第１のテーブルに関してデータを要求し受取る。たとえば、データベースサーバ１２２は、ＩＤのジョイン列で、テーブルＴ１およびＴ２において等結合を実行するよう要求を受取ってもよい。データベースサーバ１２２は、したがって、記憶システム１００からテーブルＴ１に関してデータを要求してもよい。

【００７４】
ステップ４２０において、第１のテーブルに関するデータを用いて、データベースサーバは、ジョイン演算のビルド段階を実行する－換言すれば、データベースサーバは、第１

のテーブルに対してハッシュテーブルを構築し、そのハッシュテーブルは、第１のテーブルのジョイン列に対するハッシュ関数の適用に基づいてインデックス付けされている。したがって、たとえば、データベースサーバ１２２は、各行のＩＤ列の値に基づいて各行に対してハッシュ関数を適用することにより、テーブルＴ１に対してハッシュテーブルＨ１を構築してもよい。

【００７５】

ステップ４３０において、データベースサーバはハッシュテーブルに基づいてブルームフィルタを生成する。このステップは、背景部にて論じたような技術に従って実行してもよい。たとえば、データベースサーバ１２２はブルームフィルタＢＦをハッシュテーブルＨ１に基づいて生成してもよい。

【００７６】

ステップ４４０において、データベースサーバは、ブルームフィルタを、１つ以上の記憶システムに対して、第２のテーブルに関するデータに対する１つ以上の要求との関連において通信してもよい。たとえば、ブルームフィルタＢＦは、ステップ２５０または３２０において記載されるようなステップを用いて、記憶サーバ１０６に送られてもよい。ブルームフィルタは、ステップ２４０または３１０のようなステップに従って、テーブルＴ２を表す生データに対する要求との関連において送られてもよい。ある実施例に従うと、データベースサーバは、ブルームフィルタを、形式ＳＹＳ＿ＯＰ＿ＢＬＯＯＭ＿ＦＩＬＴＥＲ（ＢＦ，Ｃｌ，Ｃ２…）の「ブルームフィルタ述語」にカプセル化し、そこにおいて、ＢＦはブルームフィルタのビットベクトル表現であり、Ｃ１，Ｃ２…はジョイン列を識別する。記憶サーバは、次いで、この述語をブルームフィルタをカプセル化するとして認識するための論理とともに構成される。

【００７７】

ステップ４５０において、記憶サーバは、ブルームフィルタを利用して、図３のステップ３４０のようなステップを用いて、第２のテーブルに対応するデータ単位を前走査およびフィルタ処理する。たとえば、記憶サーバ１０６は、図３のステップ３３０に従って、記憶装置１０２または１０４からテーブルＴ２に関してデータブロックを読出してもよい。記憶サーバ１０６は、次いで、テーブルＴ２に対するデータブロックを、図３のステップ３４２に従って、行の集合に変換してもよい。記憶サーバ１０６は、次いで、図３のステップ３４４に従って、ブルームフィルタを、行の集合における各行のジョイン列に対して評価してもよい。ブルームフィルタにヒットしない行はすべてテーブルＴ１と結合しないよう保証されるので、それらの行は、図３のステップ３４６に従って、行の集合から取除かれてもよい。

【００７８】

ステップ４６０において、ステップ４５０において生成されたフィルタ処理されたデータはデータベースサーバに返される。たとえば、記憶サーバ１０６は、フィルタ処理された行の集合をデータベースサーバ１２２に返してもよい。

【００７９】

ステップ４７０において、データベースサーバは、ジョイン演算の残りのステップを、フィルタ処理されたデータに基づいて実行する。たとえば、ブルームフィルタは偽陽性を産み出すかもしれないので、さらなるフィルタ処理を、データベースサーバにおいて、ハッシュテーブルを用いて、マージすべき行を識別するステップの前に行なう。他の例としては、データベースサーバ１２２は、フィルタ処理された行の集合を走査して、フィルタ処理された行の集合とＴ１の行との間における一致を識別してもよい。次いで、一致する行をマージして、結合された結果集合を形成してもよい。ジョイン演算の結果として、結合された結果集合におけるデータは、記憶システムの永続的なストレージにおいて少なくとも２つの異なるデータブロックに記憶される生データから構築される論理行を含む。

【００８０】

ある実施例では、ブルームフィルタ以外のデータを用いて、記憶システムに対するジョイン演算の局面を記述してもよい。たとえば、データベースサーバは、ブルームフィルタ

に対抗するものとして、記憶システムに対して、ステップ４２０において生成されたハッシュテーブル全体を単に通信してもよい。別の例としては、記憶システムは、第１のテーブルのジョイン列にはない値の集合を識別し、次いで、ある数の単純な述語を、記憶システムに対して、識別された値に基づいて通信してもよい。実際、第１のテーブルの特性を識別する任意のデータを、ブルームフィルタの代わりに、またはブルームフィルタに追加して、記憶システムに通信してもよく、記憶システムはこれらの特性に基づいてフィルタ処理を行なってもよい。

【００８１】

ある実施例では、ステップ４２０および４３０においてハッシュテーブルおよびブルームフィルタを生成するために用いられる特定のハッシュ関数は、データベースシステムのインフラストラクチャに基づいて選択されてもよい。たとえば、ハッシュテーブルおよびブルームフィルタのサイズは、公知のメモリまたは帯域幅制約に基づいて最適化されてもよい。または、用いられるハッシュ関数のタイプは、データベースシステムのインフラストラクチャに具体的に適合されてもよい。

【００８２】

ある実施例では、ブルームフィルタ（またはジョイン演算の局面を示す任意の他のデータ）は、しかしながら、記憶システムにおいて評価される複数の基準の１つであってもよい。たとえば、ステップ４４０において、ブルームフィルタに加えて、データベースサーバは、ジョイン演算に対して述べられた追加の述語を指定するメタデータを通信してもよい。データベースサーバは、たとえば、

【００８３】

【数２】

```
SELECT *
FROM   employee
    INNER JOIN department
      ON employee.DepartmentID = department.DepartmentID
       WHERE employee.DepartmentID > 1 AND
       employee.status IN ('full-time','leave')
```

【００８４】

のようなＳＱＬ文に応答して、ジョイン演算を開始していてもよい。したがって、ステップ４４０において、データベースサーバは、両方の述語“employee.DepartmentID > 1”および“employee.status IN(‘full-time', ‘leave')を示すメタデータと並んで、department（部署）テーブルに基づいてブルームフィルタを含むメタデータを送ってもよい。記憶システムは、次いで、検索されたデータブロックを、ブルームフィルタおよび双方の述語に基づいてフィルタ処理してもよい。

【００８５】

４．０．他の実現例

４．１．ブルームフィルタを生成する／送り出すべきかどうかを判断する

ある実施例に従うと、データベースサーバは、常に、ブルームフィルタを生成し、および／またはブルームフィルタ（もしくは他のジョイン条件）を記憶サーバに送るというわけではない。関与するテーブルのサイズ、ジョイン列におけるデータのタイプ、およびジョイン列における別個の値の数によっては、ハッシュテーブルおよび／またはブルームフィルタを生成することなくジョイン演算を実行するほうが、より効率がよいかもしれない。たとえば、ハッシュテーブルまたはブルームフィルタを生成することは、望ましくないメモリおよび／または処理時間量を必要とするかもしれない。別の例として、たいていの述語とは違い、ブルームフィルタ述語は特に大きいかもしれない。したがって、ブルームフィルタの記憶システムへの送信は、Ｉ／Ｏ帯域幅に関し、フィルタ処理されない生データを検索することに比べて高価であるかもしれない。

【００８６】
データベースサーバは、任意の所与のジョイン演算に関して、ブルームフィルタを生成することがブルームフィルタなしで探査することよりも効率がよいかどうかを推定するよう原価計算関数を実行するように既に構成されていてもよい。ある実施例に従うと、ブルームフィルタを記憶システムに送り出すことから結果として生ずるパフォーマンスに対する影響を考慮するよう、これらの原価計算関数を微調整しなければならない。ある状況では、データベースサーバにおいて生成し評価するのに非効率的であろうブルームフィルタが、実際には、記憶システムに対して送り出される際に利益を産み出し、というのも、記憶システムは、ブルームフィルタに基づいてデータを前フィルタ処理することにより、帯域幅消費を大きく低減することができるかもしれないからである。他の状況では、データベースサーバは、ブルームフィルタの適用はデータベースサーバにおいてのほうがより利益があるであろうと推定するかもしれない。したがって、データベースサーバは、ブルームフィルタを生成するであろうが、ブルームフィルタを記憶システムに送り出しはしないであろう。
【００８７】
ある実施例では、同様の処理を用いて、ハッシュテーブルを生成するか、ハッシュテーブルを記憶システムに送るか、または任意の他のジョイン基準を記憶システムに送るかを判断してもよい。
【００８８】
コスト関数の実現そのものは環境ごとに変動してもよい。たとえば、帯域幅の利用可能性、データベースサーバおよび記憶システムにおける計算能力、永続的記憶速度などによって、記憶システムおよび／またはデータベースサーバにおける資源の使用の認識されるコストに比較しての、帯域幅消費を低減することの認識される利益は、しばしば異なることになる。
【００８９】
４．２．ブルームフィルタをわずか１回だけ送り出す
ある実施例に従うと、データベースサーバは、ジョイン演算の第２のテーブルに関するデータを、単一のＩ／Ｏ要求に対抗するものとしての、複数のＩ／Ｏ要求を介して、１つ以上の記憶システムから検索する。ジョイン基準を複数のＩ／Ｏ要求の各々とともに送ることを回避するため、データベースサーバはジョイン基準をたった１回だけ通信してもよい。１つ以上の記憶システムは、ジョイン基準を、複数の要求の各他の要求に対して記憶および再使用してもよい。ブルームフィルタまたはハッシュテーブルなどのあるジョイン基準はおそらくはサイズが非常に大きいかもしれないため、この技術は多くの場合において複数のＩ／Ｏ要求による帯域幅消費を大きく減ずることになる。
【００９０】
たとえば、サイズ制約のため、データベースサーバは、第２のテーブルに関するデータを、複数のＩ／Ｏ要求にわたってフェッチするよう判断するかもしれない。第１のＩ／Ｏ要求前またはそれとともに、データベースサーバは第１のテーブルに関してブルームフィルタを送ってもよい。データベースサーバは、ブルームフィルタを、識別子を用いて、あるコンテキストに関連付けてもよい。記憶システムは、ブルームフィルタおよびコンテキスト識別子をキャッシュしてもよい。次いで、複数のＩ／Ｏ要求の各々は同じコンテキスト識別子を与えてもよい。Ｉ／Ｏ要求とともに送られるコンテキスト識別子を認識すると、記憶サーバは、キャッシュされたブルームフィルタを見つけ出し、それを、Ｉ／Ｏ要求に基づいて検索される任意のデータブロックに適用してもよい。
【００９１】
別の例として、データベースサーバは、あるテーブルに関してデータブロックをミラーリングする複数の記憶システムに依拠してもよい。データベースは、生データの異なる部分を、異なる記憶システムから並列で要求し、生データをより迅速に検索してもよい。ブルームフィルタを記憶システムの各々に通信する代わりに、データベースサーバは、ブルームフィルタを、（共有されるメモリまたは指定される記憶システムなどのような）記憶

システムのすべてにとってアクセス可能な位置にプッシュしてもよい。先の例におけるように、ブルームフィルタは、次いで、コンテキスト識別子を用いて、複数の記憶システムによって参照されアクセスされてもよい。

【００９２】

さらに別の例として、ジョイン演算を複数のデータベースサーバ間に分散させてもよい。ジョイン演算に対するあるコンテキストは、ジョイン演算全体に対して、データベースサーバ間で共有されるコンテキスト識別子とともに確立されてもよい。したがって、１つのデータベースサーバがブルームフィルタを生成する場合、そのブルームフィルタは、Ｉ／Ｏ要求において、データベースサーバのすべてから参照されてもよく、データベースサーバはブルームフィルタを再生成する必要がない。

【００９３】

４．３．ジョイン演算全体を記憶サーバに送り出す

ある実施例に従うと、ジョイン演算のさらなる局面は、さらに、複数の記憶システムに送り出されてもよい。たとえば、記憶サーバは、図４のステップのすべてまたはほとんどを担ってもよい。この目的のため、データベースサーバは、記憶サーバに対し、第１のテーブルおよびジョイン列のアイデンティティなどのような、ジョイン演算についてのさらなる情報を送ってもよい。この情報を用いて、記憶サーバは、ビルド段階のさまざまな局面を実行し、および／または結合された行集合を生成してもよい。ある実施例では、複数の記憶サーバはジョイン演算のこれらの局面に参加してもよく、ハッシュテーブルまたはブルームフィルタなどのような情報は複数の記憶システム間で共有されてもよい。

【００９４】

４．４．前フィルタ処理に参加する能力がない記憶システムに対処する

ある実施例に従うと、データベースサーバによって依拠される記憶システムの少なくとも１つは、いくつかの、またはすべての前フィルタ処理演算に参加する必要があるコンポーネントまたはサブコンポーネントを欠いているかもしれない。たとえば、データベースサーバは、記憶システム１００およびこれまでの記憶システムの双方に依拠するかもしれない。したがって、データベースサーバは、所与の記憶システムがある特定のフィルタ処理演算を記憶システムに送る前にその演算をサポートするかどうかを判断するよう構成される。そのような判断は、たとえば、記憶システムとの最初の交換において獲得される記憶された構成情報に基づいて達成されてもよい。この判断に基づいて、データベースサーバは、エンハンストＩ／Ｏ要求を送るべきか、およびエンハンストＩ／Ｏ要求を誰に送るべきかの双方を判断してもよい。たとえば、データベースサーバは、前フィルタ処理からは多くを得そうにない演算に関してデータを要求するときには従来の記憶アレイを優先し、他の演算に関してデータを要求するときには記憶システム１００を優先するよう構成されてもよい。

【００９５】

４．５．３つ以上のテーブル間におけるジョイン

データベースサーバは、しばしば、２つより多いテーブル間でジョインを実行するよう求められる。そのようなジョイン演算は、内部的に、一連の「ネスト化された」ジョインとして実現されてもよい。たとえば、以下のＳＱＬ文は、テーブル１、テーブル２およびテーブル３の３つのテーブルに対するジョイン演算を指定する。

【００９６】

【数３】

```
SELECT *
FROM   table1
    INNER JOIN table2
      ON table1.primarykey = table2.foreignkey
            JOIN table 3
       ON table 2.primarykey = table 3.foreignkey
```

【００９７】

データベースサーバは、まず、テーブルのうちの２つを結合する（たとえばテーブル２をテーブル３に結合する）ことによって、そのようなジョインを実行してもよい。このジョインの積を、次いで、最後のテーブルと結合する（たとえばテーブル２とテーブル３との和集合をテーブル１と結合する）。データベースサーバがブルームフィルタに依拠する場合には、ブルームフィルタは、第１のジョイン演算（たとえばテーブル３に対して適用されるべく、テーブル２に対するブルームフィルタ）および第２のジョイン演算（たとえばテーブル２とテーブル３との和集合に対して適用されるべく、テーブル１に対するブルームフィルタ）の双方に対して生成されることになる。

【００９８】

ある実施例では、２つより多いテーブル間においてジョインを処理する際、データベースサーバは、要求されたテーブルが直接結合されつつあるテーブルに対するブルームフィルタのみならず、即時ジョイン演算の積が結合されることになるテーブルに対するブルームフィルタにも基づいて、要求されたテーブルに関する生データを、記憶システムに前フィルタ処理させる。たとえば、テーブル３がテーブル２に対するブルームフィルタＢＦ２に基づいて探査されることになり、テーブル３とテーブル２との積がテーブル１に対するブルームフィルタＢＦ１に基づいて探査されることになる場合、データベースサーバは、テーブル３のデータブロックに対する適用のために、ＢＦ１およびＢＦ２の両方を記憶システムに送り出してもよい。

【００９９】

ある実施例では、即時ジョイン演算におけるテーブルの両方に関する生データを、即時ジョイン演算の結果の積がその後結合されることになるテーブルに対するブルームフィルタを用いてフィルタ処理してもよい。したがって、上記の場合においては、データベースサーバは、テーブル２に対するブルームフィルタの生成前にテーブル２のデータブロックに対する適用のためにＢＦ１を記憶システムに送り出してもよい。

【０１００】

５．０．ハードウェア概要

ある実施例に従うと、ここに記載される技術は１つ以上の特殊用途計算装置によって実現される。それら特殊用途計算装置は、それらの技術を実行するようハードワイヤードであってもよく、またはそれらの技術を実行するよう永続的にプログラミングされる１つ以上のアプリケーション特化集積回路（ＡＳＩＣ）またはフィールドプログラマブルゲートアレイ（ＦＰＧＡ）などのようなデジタル電子装置を含んでもよく、またはファームウェア、メモリ、他のストレージもしくは組合せにおけるプログラム命令に従ってそれらの技術を実行するようプログラミングされる１つ以上の汎用ハードウェアプロセッサを含んでもよい。そのような特殊用途計算装置は、さらに、それらの技術を達成するようカスタムプログラミングを伴う、カスタムハードワイヤード論理、ＡＳＩＣまたはＦＰＧＡを組合わせてもよい。特殊用途計算装置は、デスクトップコンピュータシステム、ポータブルコンピュータシステム、携帯型装置、ネットワーク化装置、またはそれらの技術を実現するようハードワイヤードおよび／またはプログラム論理を組込む任意の他の装置であってもよい。

【０１０１】

たとえば、図５は、この発明の実施例が実現されてもよいコンピュータシステム５００を例示するブロック図である。コンピュータシステム５００は、情報をやりとりするためのバス５０２または他の通信機構と、バス５０２に結合され、情報を処理するためのハードワイヤプロセッサ５０４とを含む。ハードワイヤプロセッサ５０４はたとえば汎用マイクロプロセッサであってもよい。

【０１０２】

コンピュータシステム５００は、バス５０２に結合され、プロセッサ５０４によって実行されるべき命令および情報を記憶するための、ランダムアクセスメモリ（ＲＡＭ）または他のダイナミック記憶装置などの主メモリ５０６も含む。主メモリ５０６は、プロセッサ５０４によって実行されるべき命令の実行中に一時的変数または他の中間情報を記憶するためにも使用されてよい。そのような命令は、プロセッサ５０４にとってアクセス可能な記憶媒体に記憶されると、コンピュータシステム５００を、それらの命令に指定される演算を実行するようカスタマイズされる特殊用途マシンにする。

【０１０３】

コンピュータシステム５００は、さらに、バス５０２に結合され、プロセッサ５０４のために静的情報および命令を記憶するためのリードオンリメモリ（ＲＯＭ）５０８または他の静的記憶装置を含む。磁気ディスクまたは光ディスクなどの記憶装置５１０が、情報および命令を記憶するために設けられバス５０２に結合される。

【０１０４】

コンピュータシステム５００は、コンピュータユーザに情報を表示するための、陰極線管（ＣＲＴ）などのディスプレイ５１２にバス５０２を介して結合されてもよい。アルファベット数字および他のキーを含む入力装置５１４が、プロセッサ５０４に情報およびコマンド選択を通信するためにバス５０２に結合される。他のタイプのユーザ入力装置は、マウス、トラックボールまたはカーソル方向キーなどのカーソル制御５１６であり、プロセッサ５０４に方向情報およびコマンド選択を通信し、かつディスプレイ５１２上でカーソル移動を制御する。この入力装置は、典型的には、装置が平面における位置を特定することを可能にする、第１の軸（たとえばｘ）および第２の軸（たとえばｙ）の２つの軸において２つの自由度を有する。

【０１０５】

コンピュータシステム５００は、コンピュータシステムとの組合せでコンピュータシステム５００を特殊用途マシンにするかまたはプログラムする、カスタマイズされたハードワイヤード論理、１つ以上のＡＳＩＣもしくはＦＰＧＡ、ファームウェア、および／またはプログラム論理を用いて、ここに記載される技術を実現してもよい。ある実施例に従うと、ここに記載される技術は、プロセッサ５０４が主メモリ５０６に含まれる１つ以上の命令の１つ以上のシーケンスを実行すること応答して、コンピュータシステム５００により実行される。そのような命令は、記憶装置５１０などの別の記憶媒体から主メモリ５０６に読込まれてもよい。主メモリ５０６に含まれる命令のシーケンスの実行により、プロセッサ５０４は、ここに記載された処理ステップを行なう。代替的実施例では、ソフトウェア命令の代わりに、またはこれと組合わせて、ハードワイヤード回路系が用いられてもよい。

【０１０６】

ここで用いられる「記憶媒体」という用語は、マシンをある特定の態様で動作させるデータおよび／または命令を記憶する任意の媒体のことを指す。そのような媒体は、不揮発性媒体および／または揮発性媒体を含み得る。不揮発性媒体は、たとえば、記憶装置５１０などの光ディスクまたは磁気ディスクを含む。揮発性媒体は、主メモリ５０６などのダイナミックメモリを含む。記憶媒体の一般的な形態は、たとえば、フロッピー（登録商標）ディスク、フレキシブルディスク、ハードディスク、ソリッドステートドライブ、磁気テープ、または任意の他の磁気データ記憶媒体、ＣＤ－ＲＯＭ、任意の他の光学データ記憶媒体、孔のパターンを備える任意の物理的媒体、ＲＡＭ、ＰＲＯＭ、およびＥＰＲＯＭ、ＦＬＡＳＨ－ＥＰＲＯＭ、ＮＶＲＡＭ，任意の他のメモリチップまたはカートリッジを

含む。

【０１０７】

記憶媒体は、伝送媒体とは区別されるが、それとの関連で用いられてもよい。伝送媒体は、情報を記憶媒体間で転送することに加わる。たとえば、伝送媒体は、バス５０２を含むワイヤを含む、同軸ケーブル、銅線および光ファイバを含む。伝送媒体は、無線通信および赤外線データ通信の際生成されるものなど、音波または光波の形もとり得る。

【０１０８】

さまざまな形態の媒体が、１つ以上の命令の１つ以上のシーケンスをプロセッサ５０４に搬送し実行するのに関係し得る。たとえば、命令は、最初に遠隔コンピュータの磁気ディスクまたはソリッドステートドライブ上に担持され得る。遠隔コンピュータは、命令をそのダイナミックメモリにロードし、モデムを用いて電話線を介して命令を送信することができる。コンピュータシステム５００に局在するモデムは、電話線でデータを受取り、赤外線送信機を用いてデータを赤外線信号に変換することができる。赤外線検出器が、赤外線信号で搬送されるデータを受信することができ、適切な回路系が、データをバス５０２に与えることができる。バス５０２は、データを主メモリ５０６に搬送し、プロセッサ５０４は命令をそこから検索し実行する。主メモリ５０６によって受取られた命令は、任意で、プロセッサ５０４によって実行される前または後のいずれかに記憶装置５１０上に記憶されてもよい。

【０１０９】

コンピュータシステム５００は、バス５０２に結合される通信インターフェイス５１８も含む。通信インターフェイス５１８は、ローカルネットワーク５２２に接続されるネットワークリンク５２０に結合する双方向のデータ通信を提供する。たとえば、通信インターフェイス５１８は、データ通信接続を対応するタイプの電話線に与えるよう、統合サービスデジタル網（ＩＳＤＮ）カード、ケーブルモデム、衛星モデムまたはモデムであってもよい。別の例として、通信インターフェイス５１８は、データ通信接続を互換可能なローカルエリアネットワーク（ＬＡＮ）に与えるよう、ＬＡＮカードであってもよい。ワイヤレスリンクが実現されてもよい。任意のそのような実現化例において、通信インターフェイス５１８は、さまざまな種類の情報を表わすデジタルデータストリームを搬送する電気信号、電磁信号または光信号を送信および受信する。

【０１１０】

ネットワークリンク５２０は、典型的には、１つ以上のネットワークを介してデータ通信を他のデータ装置に与える。たとえば、ネットワークリンク５２０は、ローカルネットワーク５２２を介してホストコンピュータ５２４またはインターネットサービスプロバイダ（ＩＳＰ）５２６によって操作されるデータ機器への接続を与えてもよい。ＩＳＰ５２６は、次いで、現在通常「インターネット」５２８と呼ばれているワールドワイドパケットデータ通信ネットワークを介してデータ通信サービスを提供する。ローカルネットワーク５２２およびインターネット５２８はどちらも、デジタルデータストリームを搬送する電気信号、電磁信号または光信号を使用する。さまざまなネットワークを介する信号と、コンピュータシステム５００へおよびこれからデジタルデータを搬送する、ネットワークリンク５２０上および通信インターフェイス５１８を介する信号とは、伝送媒体の例示の形である。

【０１１１】

コンピュータシステム５００は、ネットワーク、ネットワークリンク５２０および通信インターフェイス５１８を介して、プログラムコードを含む、メッセージを送信しかつデータを受信することができる。インターネットの例では、サーバ５３０は、インターネット５２８、ＩＳＰ５２６、ローカルネットワーク５２２および通信インターフェイス５１８を介してアプリケーションプログラムのための要求されたコードを送信するかもしれない。

【０１１２】

受取られたコードは、受取られたときにプロセッサ５０４によって実行されてもよく、

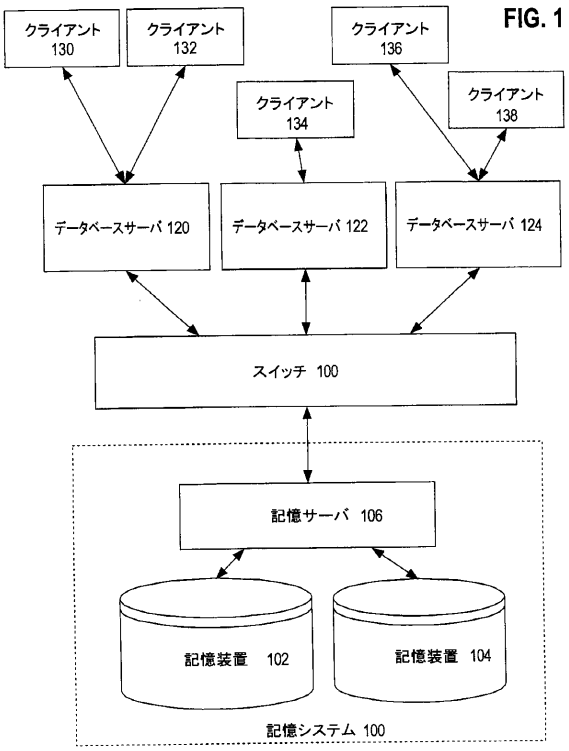
および／または後の実行のために記憶装置５１０もしくは他の不揮発性記憶部に記憶されてもよい。

【０１１３】

前述の明細書において、この発明の実施例を、実現例ごとに異なってもよい数多くの具体的な詳細を参照して記載した。したがって、この発明が何であるか、および何が出願人によってこの発明であるよう意図されるかを唯一かつ排他的に示すものは、この出願から、具体的な形で出る、任意の後の訂正を含む請求項の組である。そのような請求項に含まれる文言に対してここに明示的に述べられるすべての定義は、請求項において用いられるそのような文言の意味を支配することになる。したがって、請求項に明示的に記載されない限定、要素、特性、特徴、利点および属性は、その請求項の範囲をいかようにも限定すべきではない。明細書および図面は、したがって、限定の意味ではなく例示的な意味にみなされるべきものである。

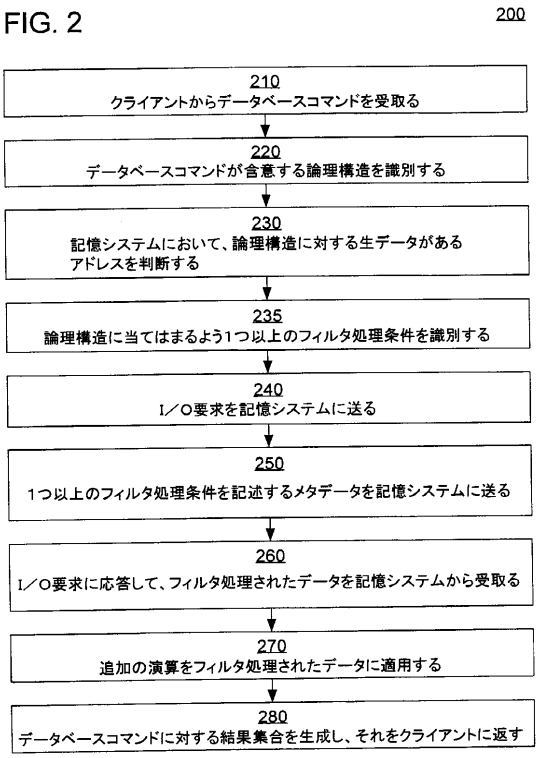
【図１】

FIG. 1



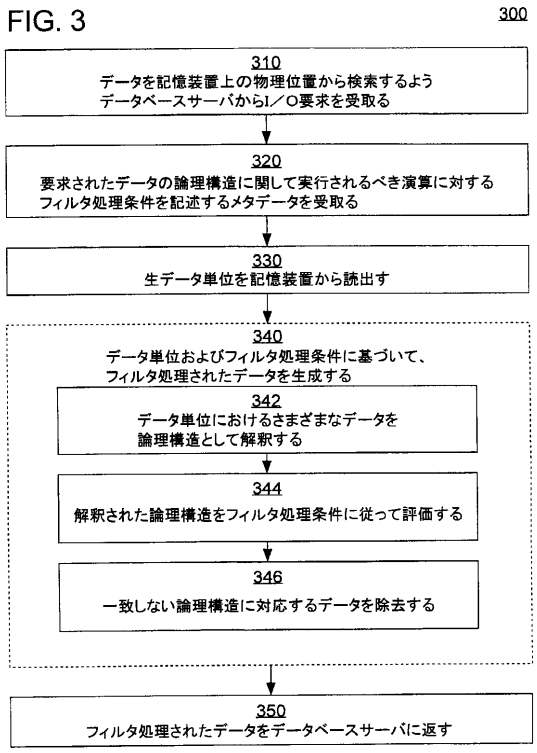
【図２】

FIG. 2



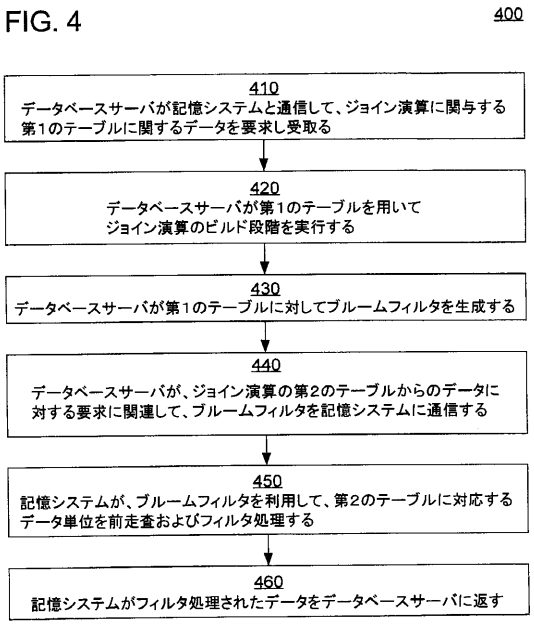
【図３】

FIG. 3



【図４】

FIG. 4



【図５】

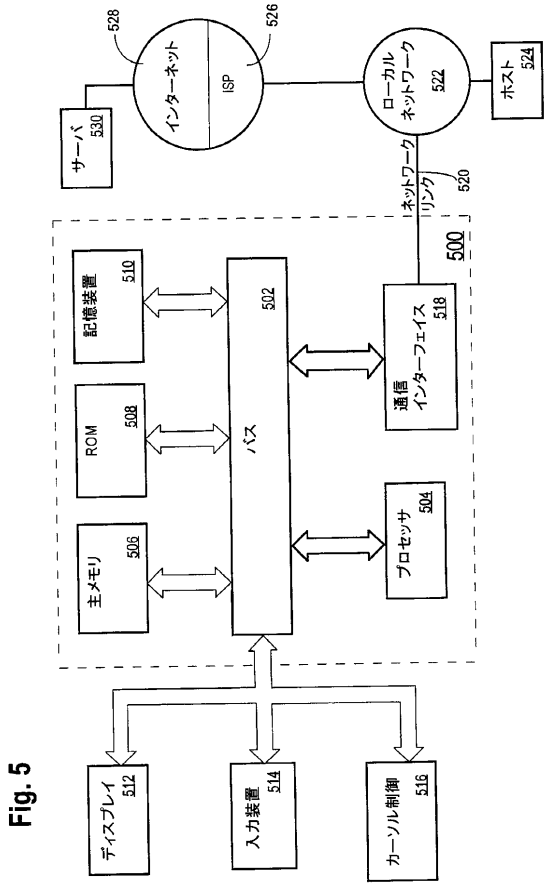


Fig. 5

フロントページの続き

(72)発明者　ラウ，イウ・ウン
アメリカ合衆国、９４５５５　カリフォルニア州、フレモント、フェアチャイルド・コモン、３４７４６
(72)発明者　ヤコブソン，ハカン
アメリカ合衆国、９４１０５　カリフォルニア州、サン・フランシスコ、ハワード・ストリート、８８、ナンバー・２００４
(72)発明者　パンチャックシャライア，ウメシュ
アメリカ合衆国、９４８０４　カリフォルニア州、リッチモンド、レークショア・コート、７９
(72)発明者　クマール，プージャン
アメリカ合衆国、９４０４０　カリフォルニア州、マウンテンビュー、コンティネンタル・サークル、７０７、ナンバー・１８２４

審査官　鹿野　博嗣

(56)参考文献　特開昭６３－２７１５２５（ＪＰ，Ａ）
武田　英昭，関係演算高速化プロセッサ，情報処理学会論文誌，日本，社団法人情報処理学会，１９９０年　８月１５日，第31巻,第8号，p.1230～1241

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｇ０６Ｆ　１７／３０