

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第4399552号
(P4399552)

(45) 発行日 平成22年1月20日 (2010. 1. 20)

(24) 登録日 平成21年11月6日 (2009. 11. 6)

| | |
|---------------------------------|-----------------------|
| (51) Int. Cl. | F I |
| G 0 6 F 12/00 (2006. 01) | G 0 6 F 12/00 5 3 3 J |
| G 0 6 F 15/00 (2006. 01) | G 0 6 F 12/00 5 1 8 A |
| | G 0 6 F 15/00 3 1 0 U |

請求項の数 30 (全 26 頁)

| | | | |
|---------------|-------------------------------|-----------|-----------------------|
| (21) 出願番号 | 特願2003-540847 (P2003-540847) | (73) 特許権者 | 502377350 |
| (86) (22) 出願日 | 平成14年11月1日 (2002. 11. 1) | | ベリサイン・インコーポレイテッド |
| (65) 公表番号 | 特表2005-510782 (P2005-510782A) | | アメリカ合衆国、カリフォルニア州 9 4 |
| (43) 公表日 | 平成17年4月21日 (2005. 4. 21) | | 0 4 3 マウンテン・ビュー、イースト・ |
| (86) 国際出願番号 | PCT/US2002/035083 | | ミドルフィールド・ロード 4 8 7 |
| (87) 国際公開番号 | W02003/038654 | (74) 代理人 | 100108453 |
| (87) 国際公開日 | 平成15年5月8日 (2003. 5. 8) | | 弁理士 村山 靖彦 |
| 審査請求日 | 平成17年9月22日 (2005. 9. 22) | (74) 代理人 | 100064908 |
| (31) 優先権主張番号 | 60/330, 842 | | 弁理士 志賀 正武 |
| (32) 優先日 | 平成13年11月1日 (2001. 11. 1) | (74) 代理人 | 100089037 |
| (33) 優先権主張国 | 米国 (US) | | 弁理士 渡邊 隆 |
| (31) 優先権主張番号 | 60/365, 169 | (74) 代理人 | 100110364 |
| (32) 優先日 | 平成14年3月19日 (2002. 3. 19) | | 弁理士 実広 信哉 |
| (33) 優先権主張国 | 米国 (US) | (74) 代理人 | 100092196 |
| | | | 弁理士 橋本 良郎 |

最終頁に続く

(54) 【発明の名称】 遠隔でデータベースを更新する方法およびシステム

(57) 【特許請求の範囲】

【請求項 1】

ネットワークを通して遠隔データベースを更新する方法において、

前記ネットワークに接続されている少なくとも1つのプロセッサと、前記プロセッサに結合されたメモリとを具備し、そのメモリは、ローカルデータベースと、前記ネットワークを通して前記遠隔データベースを更新するために前記プロセッサにより実行される命令を含んでいる、コンピュータを用いて、

ローカルデータベースに対するインクリメント変化に基づいた、それぞれ少なくとも1つのトランザクションの内容を表わしているデータを有している複数の周期的な更新データを生成し、

ネットワークを通して前記遠隔データベースへ前記複数の周期的な更新データを送信し、

前記複数の周期的な更新データを生成しながら、

初期化更新データ生成の開始時間における前記ローカルデータベースのバージョンを含んでいる初期化更新データを生成し、

前記開始時間に基づいて前記複数の周期的な更新データの中から最新の周期的な更新データを決定し、

前記開始時間に基づいて最新のトランザクションを決定し、

前記初期化更新データ、最新の周期的な更新データの識別子、最新のトランザクション識別子をネットワークを通して前記遠隔データベースへ送信するステップを含んでおり、

10

20

前記最新の周期的な更新データは、初期化更新データ生成の開始時間前に生成された周期的な更新データの内の最後に生成された周期的な更新データであり、
前記最新のトランザクションは初期化更新データ生成の開始時間前に与えられたトランザクションの内の最後のトランザクションである、方法。

【請求項 2】

前記初期化更新データを送信する前記ステップは、
前記最新の周期的な更新データの識別子を前記最新の周期的な更新データに関連付け、
前記最新のトランザクション識別子を前記最新のトランザクションに関連付けるステップを含んでいる請求項 1 記載の方法。

【請求項 3】

前記複数の周期的な更新データは前記ローカルデータベースで規則的または不規則な間隔で生成される請求項 1 記載の方法。

【請求項 4】

前記初期化更新データ生成の開始時間は、周期的な更新データの生成の開始時間と同じである請求項 1 記載の方法。

【請求項 5】

前記初期化更新データ生成の開始時間は、周期的な更新データの生成の開始時間よりも後である請求項 1 記載の方法。

【請求項 6】

周期的な更新データは複数のトランザクションの内容を表わしているデータを含んでおり、複数のトランザクションのそれぞれは特有のトランザクション識別子を有している請求項 1 記載の方法。

【請求項 7】

ネットワークを通して遠隔データベースを更新する方法において、
ネットワークに接続された少なくとも 1 つのプロセッサと、前記プロセッサに結合されたメモリとを具備し、そのメモリは、前記遠隔データベースと、前記ネットワークを通して前記遠隔データベースを更新するために前記プロセッサにより実行される命令を含んでいる、コンピュータを用いて、

ネットワークを通してローカルデータベースに対するインクリメント変化に基づいた、それぞれ少なくとも 1 つのトランザクションの内容を表わしているデータを有している複数の周期的な更新データを受信し、

前記ネットワークを通して、初期化更新データ生成の開始時間におけるローカルデータベースのバージョンを含んでいる初期化更新データを受信し、

前記初期化更新データから最新の周期的な更新データの識別子を読み取り、

前記初期化更新データから最新のトランザクション識別子を読み取り、

前記最新の周期的な更新データの識別子から最新の周期的な更新データを決定し、前記最新の周期的な更新データの決定は前記開始時間に基づいており、

前記最新のトランザクション識別子から最新のトランザクションを決定し、前記最新のトランザクションの決定は前記開始時間に基づいており、

前記最新のトランザクションの後に生成されたトランザクションを遠隔データベースへ供給し、

前記最新の周期的な更新データの後に生成された前記周期的な更新データを遠隔データベースへ供給するステップを含んでおり、

前記最新の周期的な更新データは、初期化更新データ生成の開始時間前に生成された周期的な更新データの内の最後に生成された周期的な更新データであり、
前記最新のトランザクションは初期化更新データ生成の開始時間前に与えられたトランザクションの内の最後のトランザクションである、方法。

【請求項 8】

前記最新のトランザクションを決定した後に、前記初期化更新データ生成の開始時間よりも前の時間で生成された前記周期的な更新データを廃棄するステップをさらに含んでい

10

20

30

40

50

る請求項 7 記載の方法。

【請求項 9】

前記複数の周期的な更新データは、規則的な間隔で一度に 1 つ受信される請求項 7 記載の方法。

【請求項 10】

前記複数の周期的な更新データは、周期的な間隔でバッチで受信される請求項 7 記載の方法。

【請求項 11】

周期的な更新データは複数のトランザクションの内容を表わしているデータを含み、各複数のトランザクションは特有のトランザクション識別子を有している請求項 7 記載の方法。

【請求項 12】

ネットワークを通して遠隔データベースを更新する方法において、

前記ネットワークに接続されている少なくとも 1 つのプロセッサと、前記プロセッサに結合されたメモリとを具備し、そのメモリは、ローカルデータベースと、前記ネットワークを通して前記遠隔データベースを更新するために前記プロセッサにより実行される命令を含んでいる、コンピュータを用いて、

ローカルデータベースに対するインクリメント変化に基づいた、それぞれ少なくとも 1 つのトランザクションの内容を表わしているデータを有している複数の周期的な更新データを生成し、

初期化更新データ生成の開始時間におけるローカルデータベースのバージョンを含んでいる初期化更新データと、前記開始時間前に生成された最新の周期的な更新データに関連する更新データの識別子と、前記開始時間前に与えられた最新のトランザクションに関連するトランザクション識別子とを生成し、

前記初期化更新データと、前記更新データの識別子と、前記トランザクション識別子とをネットワークを通して前記遠隔データベースへ送信するステップを含んでおり、

前記最新の周期的な更新データは、初期化更新データ生成の開始時間前に生成された周期的な更新データの内の最後に生成された周期的な更新データであり、
前記最新のトランザクションは初期化更新データ生成の開始時間前に与えられたトランザクションの内の最後のトランザクションである、方法。

【請求項 13】

前記複数の周期的な更新データは規則的な間隔で生成される請求項 12 記載の方法。

【請求項 14】

前記複数の周期的な更新データは不規則な間隔で生成される請求項 12 記載の方法。

【請求項 15】

前記初期化更新データ生成の開始時間は、周期的な更新データの生成の開始時間と同じである請求項 12 記載の方法。

【請求項 16】

前記初期化更新データ生成の開始時間は、周期的な更新データの生成の開始時間よりも後である請求項 12 記載の方法。

【請求項 17】

ネットワークを通して遠隔データベースを更新するシステムにおいて、

ネットワークに接続されている少なくとも 1 つのプロセッサと、

前記プロセッサに結合されたメモリとを具備し、

前記メモリは、ローカルデータベースと、前記ネットワークを通して前記遠隔データベースを更新する方法を行うために前記プロセッサにより実行される命令とを含んでおり、前記更新する方法は、

ローカルデータベースに対するインクリメント変化に基づいた、それぞれ少なくとも 1 つのトランザクションの内容を表わしているデータを有している複数の周期的な更新データを生成し、

10

20

30

40

50

前記ネットワークを通して前記遠隔データベースへ前記複数の周期的な更新データを送信し、

前記複数の周期的な更新データを生成しながら、

初期化更新データ生成の開始時間における前記ローカルデータベースのバージョンを含んでいる初期化更新データを生成し、

前記開始時間に基づいて前記複数の周期的な更新データの中から最新の周期的更新データを決定し、

前記開始時間に基づいて最新のトランザクションを決定し、

前記初期化更新データ、最新の周期的な更新データの識別子、最新のトランザクション識別子を前記ネットワークを通して前記遠隔データベースへ送信するステップを含んでおり、

前記最新の周期的な更新データは、初期化更新データ生成の開始時間前に生成された周期的な更新データの内の最後に生成された周期的な更新データであり、

前記最新のトランザクションは初期化更新データ生成の開始時間前に与えられたトランザクションの内の最後のトランザクションである、システム。

【請求項 18】

前記初期化更新データを送信する前記ステップは、

前記最新の周期的な更新データの識別子を前記最新の周期的な更新データに関連付け、

前記最新のトランザクション識別子を前記最新のトランザクションに関連付ける請求項 17 記載のシステム。

【請求項 19】

前記複数の周期的な更新データは前記ローカルデータベースで規則的または不規則な間隔で発生成される請求項 17 記載のシステム。

【請求項 20】

前記初期化更新データ生成の開始時間は、周期的な更新データの生成の開始時間と同じである請求項 17 記載のシステム。

【請求項 21】

前記初期化更新データ生成の開始時間は、周期的な更新データ生成の開始時間よりも後である請求項 17 記載のシステム。

【請求項 22】

周期的な更新データは複数のトランザクションの内容を表わしているデータを含んでおり、複数のトランザクションのそれぞれは特有のトランザクション識別子を有している請求項 17 記載のシステム。

【請求項 23】

ネットワークを通して遠隔データベースを更新するシステムにおいて、

ネットワークに接続される少なくとも1つのプロセッサと、

前記プロセッサに結合されたメモリとを具備し、

前記メモリは、遠隔データベースと、前記ネットワークを通して前記遠隔データベースを更新する方法を行うために前記プロセッサにより実行される命令とを含んでおり、その更新方法は、

前記ネットワークを通してローカルデータベースに対するインクリメント変化に基づいた、それぞれ少なくとも1つのトランザクションの内容を表わしているデータを有している複数の周期的な更新データを受信し、

前記ネットワークを通して、初期化更新データ生成の開始時間における前記ローカルデータベースのバージョンを含んでいる初期化更新データを受信し、

前記初期化更新データから最新の周期的な更新データの識別子を読み取り、

前記初期化更新データから最新のトランザクション識別子を読み取り、

前記最新の周期的な更新データの識別子から最新の周期的な更新データを決定し、前記最新の周期的な更新データは前記開始時間に基づいており、

前記最新のトランザクション識別子から最新のトランザクションを決定し、前記最新の

10

20

30

40

50

トランザクションは前記開始時間に基づいており、

前記最新のトランザクションの後に生成されたトランザクションを前記遠隔データベースへ供給し、

前記最新の周期的な更新データの後に生成された前記周期的な更新データを前記遠隔データベースへ供給するステップを含んでおり、

前記最新の周期的な更新データは、初期化更新データ生成の開始時間前に生成された周期的な更新データの内の最後に生成された周期的な更新データであり、

前記最新のトランザクションは初期化更新データ生成の開始時間前に与えられたトランザクションの内の最後のトランザクションである、システム。

【請求項 2 4】

さらに、前記最新のトランザクションを決定した後に、前記初期化更新データ発生を開始時間よりも前の時間で生成された前記周期的な更新データを廃棄する請求項 2 3 記載のシステム。

【請求項 2 5】

前記複数の周期的な更新データは、規則的または不規則の間隔で、一度に 1 つ受信される請求項 2 3 記載のシステム。

【請求項 2 6】

前記複数の周期的な更新データは、規則的または不規則の間隔で、バッチで受信される請求項 2 3 記載のシステム。

【請求項 2 7】

周期的な更新データは複数のトランザクションの内容を表わしているデータを含み、複数のトランザクションのそれぞれは特有のトランザクション識別子を有し、トランザクション識別子はランダムな順序である請求項 2 3 記載のシステム。

【請求項 2 8】

ネットワークを通して遠隔データベースを更新する方法をプロセッサにより実行させるプログラム命令を含んでいるマシンの読取り可能な媒体において、この方法は、

ローカルデータベースに対するインクリメント変化に基づいた、それぞれ少なくとも 1 つのトランザクションの内容を表わしているデータを有している複数の周期的な更新データを生成し、

前記ネットワークを通して前記遠隔データベースへ前記複数の周期的な更新データを送信し、

前記複数の周期的な更新データを生成しながら、

初期化更新データ生成の開始時間におけるローカルデータベースのバージョンを含んでいる初期化更新データを生成し、

前記開始時間に基づいて前記複数の周期的な更新データの中から最新の周期的な更新データを決定し、

前記開始時間に基づいて最新のトランザクションを決定し、

最新の周期的な更新データの識別子を前記最新の周期的な更新データに関連付け、

最新のトランザクション識別子を前記最新のトランザクションに関連付け、

前記初期化更新データ、前記最新の周期的な更新データの識別子、前記最新のトランザクション識別子を前記ネットワークを通して前記遠隔データベースへ送信するステップを含んでおり、

前記最新の周期的な更新データは、初期化更新データ生成の開始時間前に生成された周期的な更新データの内の最後に生成された周期的な更新データであり、

前記最新のトランザクションは初期化更新データ生成の開始時間前に与えられたトランザクションの内の最後のトランザクションである、マシンの読取り可能な媒体。

【請求項 2 9】

ネットワークを通して遠隔データベースを更新する方法をプロセッサにより実行させるプログラム命令を含んでいるマシンの読取り可能な媒体において、この方法は、

前記ネットワークを通してローカルデータベースに対するインクリメント変化に基づい

10

20

30

40

50

た、それぞれ少なくとも1つのトランザクションの内容を表わしているデータを有している複数の周期的な更新データを受信し、

前記ネットワークを通して、初期化更新データ生成の開始時間におけるローカルデータベースのバージョンを含んでいる初期化更新データを受信し、

前記初期化更新データから最新の周期的な更新データの識別子を読み取り、

前記初期化更新データから最新のトランザクション識別子を読み取り、

前記最新の周期的な更新データの識別子から最新の周期的な更新データを決定し、前記最新の周期的な更新データは前記開始時間に基づいており、

前記最新のトランザクション識別子から最新のトランザクションを決定し、前記最新のトランザクションは前記開始時間に基づいており、

前記最新のトランザクションの後に生成されたトランザクションを前記遠隔データベースへ供給し、

前記最新の周期的な更新データの後に生成された周期的な更新データを前記遠隔データベースへ供給するステップを含んでおり、

前記最新の周期的な更新データは、初期化更新データ生成の開始時間前に生成された周期的な更新データの内の最後に生成された周期的な更新データであり、

前記最新のトランザクションは初期化更新データ生成の開始時間前に与えられたトランザクションの内の最後のトランザクションである、マシンの読み取り可能な媒体。

【請求項30】

ローカルデータベースに対するインクリメント変化に基づいた、それぞれ少なくとも1つのトランザクションの内容を表わしているデータを有している複数の周期的な更新データを生成する手段と、

初期化更新データ生成の開始時間におけるローカルデータベースのバージョンを含んでいる初期化更新データを生成し、更新データの識別子を前記開始時間前に生成された最新の周期的な更新データに関連付け、トランザクション識別子を前記開始時間前に与えられた最新のトランザクションに関連付ける手段と、

前記初期化更新データと、前記更新データの識別子と、前記トランザクション識別子とをネットワークを通して遠隔データベースへ送信する手段とを含んでおり、

前記最新の周期的な更新データは、初期化更新データ生成の開始時間前に生成された周期的な更新データの内の最後に生成された周期的な更新データであり、

前記最新のトランザクションは初期化更新データ生成の開始時間前に与えられたトランザクションの内の最後のトランザクションである、更新データ生成装置。

【発明の詳細な説明】

【技術分野】

【0001】

本発明の実施形態はコンピュータデータベースに関し、特に確実にデータベースを更新する方法およびシステムに関する。

【背景技術】

【0002】

この非暫定出願は、全体を参考文献としている2001年11月1日出願の米国特許暫定出願第60/330,842号明細書と、全体を参考文献としている2002年3月19日出願の米国特許暫定出願第60/365,169号明細書の利点を主張している。

【0003】

データベースのサイズの増加と高い分配性により、ネットワーク中の関連するデータベースがデータの同一のバージョンを含んでいることを確認することが非常に困難になっている。1つのデータベースに対して大きな変化が存在するならば、他のデータベースは可能な限りこれらの変更を含むように更新される必要がある。これらの更新を行うことは更新する多量のデータ量を多数のデータベースへ頻繁に移動することを含んでいる。このようなプロセスの潜在的な複雑性は膨大なものである。

【発明の開示】

【発明が解決しようとする課題】

【0004】

この問題は、通信が確実ではないシステムではさらに大きくなる。この場合、データは転送中に失われる可能性がある。このような結果、データは再度送信され、他のデータベースは再度全て更新されなければならない。このような反復はシステムの効率と、データベースが更新データを含む程度を著しく減少させる。

【課題を解決するための手段】

【0005】

本発明の実施形態はネットワークによって遠隔データベースを確実に更新する方法およびシステムを提供する。実施形態では、ローカルデータベースに対するインクリメンタルな変化に基づいて複数の周期的更新（以後、“送信ファイル”という）が行われる。各周期的更新は少なくとも1つのトランザクションを含んでいる。開示時間のローカルデータベースのバージョンを含んでいる初期化更新（以後“送信ファイルの初期化”）が行われる。さらに、開始時間前に行われる最新の周期的な更新に関連する識別子と、開始時間前に行われた最新のトランザクションに関連する識別子が発生される。実施形態は有効に、遠隔データベースを確実に更新するため、送信ファイルの結合解除と、送信ファイルの初期化を行う。

【発明を実施するための最良の形態】

【0006】

図1は、本発明の1実施形態にしたがったシステムを示すブロック図である。一般的に、システム100は大きいメモリに内在するデータベースをホストし、サーチリクエストを受信し、サーチ応答をネットワークで提供する。例えばシステム100は例えばニューヨーク、ArmonkのInternational Business Machines社により製造されたIBM RS/6000(R) M80またはS80および、カリフォルニア、サンタクララのSun Microsystems社により製造されているSun Enterprise（商標名）1000のような対称的な多処理（SMP）コンピュータであってもよい。システム100はまた例えばカリフォルニア、Palo AltoのHewlett-Packard Companyによる製造の（2つのIntel Pentium(R) III 866 MHzプロセッサを含む）Compaq ProLiant（商標名）ML530のようなマルチプロセッサパーソナルコンピュータであってもよい。システム100は例えばIBM AIX(R) 4、Sun Solaris（商標名）8 Operating Environment、Red Hat Linux(R) 6.2等のような多処理オペレーティングシステムを含んでもよい。システム100はネットワーク124で定期的な更新を受信し、それは同時にデータベースに含まれる。本発明の実施形態はデータベースのロックまたはアクセス制御を使用せずにデータベースに各更新を含ませることによって非常に高いデータベース検索および更新を実現することができる。

【0007】

1実施形態では、システム100はバス101に結合されている少なくとも1つのプロセッサ102-1を含んでいる。プロセッサ102-1は内部メモリキャッシュ（例えば明白には示さないがL1キャッシュ）を含んでいてもよい。二次メモリキャッシュ103-1（例えばL2キャッシュ、L2/L3キャッシュ等）はプロセッサ102-1とバス101との間に存在する。好ましい実施形態では、システム100はバス101に結合された複数のプロセッサ102-1...102-Pを含んでもよい。複数の二次メモリキャッシュ103-1...103-Pもまた複数のプロセッサ102-1...102-Pとバス101（例えばルックスルーアーキテクチャ）の間に存在し、またはその代わりに少なくとも1つの二次メモリキャッシュ103-1はバス101（例えばルックアサイドアーキテクチャ）に結合されている。システム100は複数のプロセッサ102-1...102-Pにより実行される情報および命令を記録するために例えばバス101に結合されたランダムアクセスメモリ（RAM）等のようなメモリ104を含んでいる。

【0008】

メモリ104は例えばインターネットドメイン名をインターネットアドレスに変換し、名称または電話番号をネットワークアドレスへ変換し、加入者プロフィールデータを提供し更新し、ユーザの現在のデータを提供し更新するための大きいデータベースを記憶してい

10

20

30

40

50

る。データベースのサイズと、毎秒のトランザクション数の両者は非常に大きいと有効である。例えばメモリ104はRAMにおいて少なくとも64GBを含み、500M（即ち 500×10^6 ）記録ドメイン名データベース、500M記録加入者データベース、450M記録電話番号ポータブルデータベース等を含んでいる。

【0009】

例えば少なくとも64ビットバス101と64ビットメモリ104に結合されている少なくとも1つの64ビットのビッグエンディアンプロセッサ102-1を含んでいる例示的な64ビットシステムアーキテクチャでは、8バイトポインタ値は単一の割り込み可能ではない動作を使用して8バイト境界のメモリアドレス（即ち8または例えば8Nにより分割可能なメモリアドレス）に書き込まれる。通常、二次メモリキャッシュ103-1の存在は単にメモリ104への8バイトポインタの書き込みを遅延する。例えば1実施形態では、二次メモリキャッシュ103-1はライトスルーモードで動作するルックスルーキャッシュであり、それによって単一の8バイトの記憶命令は割り込みなしに、2程度の少数のシステムのクロックサイクルでプロセッサ102-1からメモリ104へデータの8バイトを移動する。別の実施形態では、二次メモリキャッシュ103-1はライトバックモードで動作するルックスルーキャッシュであり、それによって8バイトのポインタが最初に二次メモリキャッシュ103-1に書き込まれ、これはその後、例えば8バイトのポインタが記憶されるキャッシュラインがメモリ104に書き込まれるとき（即ち例えば特定のキャッシュラインまたは二次メモリキャッシュ全体が“フラッシュ”されるとき）のような後の時間に8バイトポインタをメモリ104へ書き込む。

【0010】

最終的に、プロセッサ102-1の図から、データが一度プロセッサ102-1の出力ピンにラッチされると、データの全ての8バイトは1つの隣接する割り込みされない転送でメモリ104に書き込まれ、これは存在するならば二次メモリキャッシュ103-1の効果により遅延される。プロセッサ102-2...102-Pの図から、一度データがプロセッサ102-1の出力ピンにラッチされると、データの全ての8バイトは1つの隣接する割り込みされない転送でメモリ104に書き込まれ、これは二次メモリキャッシュ103-1...103Pを横切ってキャッシュのコヒーレンシープロトコルにより強化され、存在するならばメモリ104への書き込みを遅延する。

【0011】

しかしながら、8バイトポインタ値は8バイト境界を横切るメモリアドレスのようなメモリ104中の誤整列された位置に書き込まれるならば、データの全ての8バイトは単一の8バイト記憶命令を使用してプロセッサ102-1から転送されることができない。その代わりにプロセッサ102-1は2つの別々の異なる記憶命令を発生する可能性がある。例えばメモリアドレスが8バイト境界前の4バイトで開始するならば（例えば8N-4）、第1の記憶命令は4つの上位桁バイトをメモリ104へ転送し（例えば8N-4）、第2の記憶命令は4つの最小桁バイトをメモリ104へ転送する（例えば8N）。重要なことは、これらの2つの別々の記憶命令間では、プロセッサ102-1は割り込みされず、またはプロセッサ102-1は別のシステムコンポーネント（例えばプロセッサ102-P等）へのバス101の制御を失う可能性がある。結果として、メモリ104に位置するポインタ値はプロセッサ102-1が第2の記憶命令を完了できるまで、無効である。別のコンポーネントがこのメモリ位置への単一の割り込み可能ではないメモリ読取りを開始するならば、無効な値は恐らく有効値1として戻される。

【0012】

同様に、新しい4バイトポインタ値は単一の割り込み可能ではない動作を使用して4（例えば4N）により分割可能なメモリアドレスへ書き込まれる。前述の例では、4バイトポインタ値は単一の記憶命令を使用して8N-4メモリ位置に書き込まれることに注意する。勿論、4バイトポインタ値が4バイト境界を横切る位置、例えば4N-2へ書き込まれるならば、データの全ての4バイトは単一の記憶命令を使用してプロセッサ102-1から転送されることができず、メモリ104に存在するポインタ値は幾らかの時間期間で無効であ

る。

【 0 0 1 3 】

システム100はまたプロセッサ102-1に対する静的情報および命令を記憶するためバス101に結合された読取り専用メモリ (R O M) 106、または他の静的記憶装置を含んでいる。磁気または光ディスクのような記憶装置108は情報および命令を記憶するためバス101に結合されている。システム100はまたバス101に結合されているディスプレイ110 (例えば L C D モニタ) および入力装置112 (例えばキーボード、マウス、トラックボール等) を含んでいる。システム100は複数のネットワークインターフェース114-1...114-Qを含んでおり、これは種々のタイプの情報を表すデジタルデータストリームを有する電気、電磁または光信号を送信し受信する。1実施形態では、ネットワークインターフェース114-1はバス101と構内網 (L A N) 122に結合され、一方、ネットワークインターフェース114-0はバス101と広域網 (W A N) 124に結合されている。複数のネットワークインターフェース114-1...114-0は例えばギガビットイーサネット (R) (例えば2002年出版の I E E E 標準 802.3-2002)、ファイバチャネル (例えば1994年出版の A N S I 標準 X.3230-1994) 等を含んでいる種々のネットワークプロトコルをサポートする。複数のネットワークコンピュータ120-1...120-Nは L A N 122と W A N 124に結合されている。1実施形態では L A N 122と W A N 124は物理的に異なったネットワークでもよく、一方別の実施形態では L A N 122と W A N 124はネットワークゲートウェイまたはルータ (明白には示さない) を介している。代わりに、 L A N 122と W A N 124は同一のネットワークであってもよい。

【 0 0 1 4 】

前述したように、システム100は D N S 解決サービスを提供してもよい。D N S 解決実施形態では、D N S 解決サービスは通常ネットワーク転送とルックアップ機能との間で分割されている。例えばシステム100は大きいデータセットのデータルックアップに最適化されているバックエンドルックアップエンジン (L U E) であり、一方複数のネットワークコンピュータ120-1...120Nはネットワーク処理と転送に最適化された複数のフロントエンドプロトコルエンジン (P E) である。L U E は高速度で高スループットのサーチおよび更新を促すためメモリ104中に D N S 記録セット全体を記憶する強力なマルチプロセッササーバである。代替の実施形態では、D N S 解決サービスは一連の強力なマイクロプロセッササーバまたは L U E により提供され、それぞれ高速度で高スループットのサーチおよび更新を促すためにメモリに D N S 記録セット全体のサブセットを記憶する。

【 0 0 1 5 】

反対に、複数の P E は一般的な低プロファイルで P C ベースの効率的にマルチタスクオペレーティングシステムを動作するマシン (例えば Red Hat Linux (R) 6.2) であり、D N S 解決のために利用可能なリソースを最大にするために L U E におけるネットワーク処理転送ロードを最少にする。P E はワイヤライン D N S プロトコルのニュアンスを管理し、L A N 122によって L U E に対する無効な D N S の問い合わせと多重の有効な D N S の問い合わせに回答する。D N S 記録サブセットを記憶する多数の L U E を含む代替の実施形態では、P E は適切な L U E に対する各有効な D N S 問い合わせと多重の有効な D N S 問い合わせを受信する L U E を決定する。単一の L U E に対する P E 数は、例えば毎秒処理される D N S 問い合わせ数と、特定のシステムの性能特性によって決定される。他の計算も適切なマッピング比および動作態様の決定に使用されてもよい。

【 0 0 1 6 】

通常、例えば電話機番号の解決、S S 7 シグナリング処理、地上位置決定、電話番号と加入者のマッピング、加入者位置、存在決定等を含む他の大容量の問い合わせベースの実施形態がサポートされる。

【 0 0 1 7 】

1実施形態では、中央オンライントランザクション処理 (O L T P) サーバ140-1は W A N 124に結合され、データベース142-1に対する付加、変更および消去 (即ち更新トラフィック) を種々のソースから受信する。データベース142-1のローカルコピーを含んでいる O L T P サーバ140-1は W A N 124によってシステム100へ更新を送信する。O L T P サ

10

20

30

40

50

サーバ140-1はHyper Text Transmission Protocol (HTTP)、Registry Registrar Protocol (RRP)、Extensible Provisioning Protocol (EPP)、Service Management System/800 Mechanized Generic Interface (MGI)およびその他のオンライン規定プロトコルを含む種々のフォーマットおよびプロトコルでの更新トラフィックの処理に最適化されている。読み取り専用LUEのコンステレーションはOLTPサーバ140-1からの高容量のインクリメント更新と結合する高速度サーチ能力を提供するためにハブおよびスポークアーキテクチャに配備される。

【0018】

代替の実施形態では、データは多数のOLTPサーバ140-1...140-Sにわたって分配され、それぞれWAN124に結合される。OLTPサーバ140-1...140-Sは種々のソースからそれらのそれぞれのデータベース142-1...142-S（明白には図示せず）に対する付加、変更、消去（即ち更新トラフィック）を受信する。データベース142-1...142-Sのコピーと他のダイナミックに生成されたデータ等を含んでもよいOLTPサーバ140-1...140-SはWAN124によってシステム100へ更新を送信する。例えば、地上位置の実施形態では、OLTPサーバ140-1...140-Sは遠隔センサのグループから更新トラフィックを受信する。別の代替の実施形態では、複数のネットワークコンピュータ120-1...120-NもまたWAN124またはLAN122にわたって種々のソースから付加、変更、消去（即ち更新トラフィック）を受信する。この実施形態では、複数のネットワークコンピュータ120-1...120-Nは問い合わせと同様に更新をシステム100へ送信する。

【0019】

DNS解決実施形態では、各PE（例えば複数のネットワークコンピュータ120-1...120-N）は広域網（例えばWAN124）で受信された幾つかのDNS問い合わせメッセージを単一のリクエストスーパーパケットへ結合または多重化し、リクエストスーパーパケットを構内網（例えばLAN122）でLUE（例えばシステム100）へ送信する。LUEは幾つかのDNS問い合わせメッセージ応答を単一の応答スーパーパケットへ結合または多重化し、応答スーパーパケットを構内網で適切なPEへ送信する。通常、リクエストまたは応答スーパーパケットの最大のサイズは物理的なネットワーク層の最大の送信単位MTU（例えばギガビットイーサネット（R））により限定される。例えば100バイトまたは200バイトよりも小さい典型的なDNS問い合わせおよび応答メッセージのサイズはそれぞれ30を超える問い合わせが単一のリクエストスーパーパケットへ多重化されることを可能にし、15を超える応答が単一の応答スーパーパケットに多重化されることを可能にする。しかしながら、さらに少数の問い合わせ（例えば20の問い合わせ）が応答（例えば10応答）のMTUオーバーフローを避けるために単一のリクエストスーパーパケットに含まれる。さらに大きいMTUサイズでは、多重化された問い合わせおよび応答数はしたがって増加されてもよい。

【0020】

各マルチタスクPEはそれぞれDNS問い合わせおよび応答を管理するため、インバウンドスレッドとアウトバウンドスレッドを含んでいる。例えばインバウンドスレッドは広域網で受信された入来するDNS問い合わせパケットからDNS問い合わせコンポーネントの整列を解除し、問合せの数ミリ秒を単一のリクエストスーパーパケットへ多重化する。インバウンドスレッドはその後、構内網によってリクエストスーパーパケットをLUEへ送信する。反対に、アウトバウンドスレッドはLUEから応答スーパーパケットを受信し、そこに含まれる応答をデマルチプレクスし、種々のフィールドを有効なDNS応答へ整列し、これはその後、広域網で送信される。一般に、前述するように、他の大容量の問い合わせベースの実施形態がサポートされる。

【0021】

1実施形態では、リクエストスーパーパケットはまた例えばソースアドレス、プロトコルタイプ等のような各DNS問い合わせに関連する状態情報を含んでいる。LUEは状態情報と、関連されるDNS応答を応答スーパーパケット内に含んでいる。各PEはその後、LUEから送信された情報を使用して有効なDNS応答メッセージを構成し戻す。結果

として、各 P E は状態のないマシンとして有効に動作し、即ち有効な D N S 応答は応答スーパーパケットに含まれる情報から形成される。通常、L U E は入来するスーパーパケットが発生された P E へ応答スーパーパケットを戻し、他の変更が可能であることは明らかである。

【 0 0 2 2 】

別の実施形態では、各 P E は各 D N S 問い合わせに関連する状態情報を維持し、リクエストスーパーパケット内の状態情報に対する基準を含み、または処理する。L U E は状態情報の基準と、関連する D N S 応答を応答スーパーパケット内に含んでいる。各 P E はその後、L U E から送信される状態情報の基準と、そこに維持された状態情報を使用して有効な D N S 応答メッセージを構成し戻す。この実施形態では、L U E は入来するスーパーパケットが発生された P E へ応答スーパーパケットを戻す。

10

【 0 0 2 3 】

図 2 は、本発明の 1 実施形態にしたがったハブおよびスポークアーキテクチャのブロック図である。通常システムは（中央 O L T P ハブ 140 に含まれていてもよい）ローカルデータベース 200 と、例えばインターネットまたは L A N 122 のような任意の接続機構によりローカルデータベース 200 に接続されている（L U E 100 に含まれていてもよい）1 以上の遠隔データベース 210 を含んでいる。データベースは更新データを送信し、受信する。

【 0 0 2 4 】

図 3 を参照すると、本発明の 1 実施形態で、ローカルデータベース 200 は遠隔データベース 210 を更新するために遠隔データベース 210 へ F 送信ファイル 300-1...300-F と、初期化送信ファイル 310 を送信する。更新ファイルは、多数の送信ファイル 300 と 1 つの送信ファイル 300 と 1 つの初期化送信ファイル 310、多数の送信ファイル 300 と 1 つの初期化送信ファイル 310、送信ファイル 300 単独、または初期化送信ファイル 310 単独のように、個別にまたはバッチで送信されることができる。

20

【 0 0 2 5 】

本発明の 1 実施形態では、プロセッサ 104 はローカルデータベース 200 から更新データを含んでいる送信ファイル 300 および / または初期化送信ファイル 310 を受信する。システム 150 は送信ファイル 300 と初期化送信ファイル 310 を通信インターフェース 118 を介して遠隔データベース 210 で受信する。プロセッサ 104 はその後、送信ファイル 300 または初期化送信ファイル 310 の更新データを遠隔データベース 210 の対応するデータと比較する。データが遠隔データベース 210 で異なっているならば、プロセッサ 104 は送信ファイル 300 または初期化送信ファイル 310 を遠隔データベース 210 へ供給する。したがって遠隔データベース 210 は結果としてローカルデータベース 200 中の更新データと一致する更新されたデータを有することができる。

30

【 0 0 2 6 】

図 4 は本発明の 1 実施形態にしたがった送信ファイル 300 を示している。ファイル 300 のフィールドは例えばファイル識別子 400、ファイル発生時間 402、ファイル中のトランザクション数 N 404、ファイルの全サイズ 406、チェックサムまたは任意のこのようなエラーチェックインジケータ 408、（トランザクション識別子を含んでいる）トランザクション 410-1...410-N を含んでいる。これらの送信ファイルフィールドは例示であり、本発明の実施形態の技術的範囲を限定することを目的とするものではない。任意の有効なフィールドは送信ファイル 300 に含まれることができる。

40

【 0 0 2 7 】

送信ファイル 300 は時間における 2 つのポイント間のローカルデータベース 200 に対する変更を含んでいる。これらの変更は例えば新しい識別子（即ちデータ記録の識別子）の付加、既存の識別子の消去、識別子に関連する 1 以上のデータ記録の変更、識別子の名称変更、n o - o p 等を含んでいる。1 以上のこれらの変化は順番に行われ、トランザクションと呼ばれる、送信ファイル 300 はこれらのトランザクションの特有の識別子を含んでいる。トランザクションはこれらがローカルデータベース 200 で行われるように送信ファイル 300 に記録される。さらに 1 よりも多数の変化を含んでいるトランザクションに対して

50

は、変更はこれらがローカルデータベース200で行われる順序でトランザクション内で記録される。

【 0 0 2 8 】

一般に、トランザクション識別子は任意の順序でトランザクションに割当てられる。即ち、トランザクション識別子は時間にわたって単調に増加する必要はない。例えば、2つの連続的なトランザクションは1 0 0 0 2が後続する1 0 0 0 4のトランザクション識別子を有する。したがってトランザクションが発生される順序は現在のファイル300-F中の位置付けまたは先行するファイル300-(F-1)の位置により決定される。通常、トランザクションは、1つの送信ファイルのアプリケーション内で遠隔データベースの更新を十分に完成するため隣接ファイル300の範囲を含めない。これはネットワーク遅延による更新の中断を防止し、これは遠隔データベース210でエラーのあるデータを生じる可能性がある。

【 0 0 2 9 】

図5は、本発明の1実施形態にしたがった初期化送信ファイル310を示している。初期化送信ファイル310のフィールドは例えばファイル識別子500、ファイル発生時間502、ファイル中のトランザクションの数N504、ファイルの全サイズ506、チェックサムまたは任意のこのようなエラーチェックインジケータ508、ローカルデータベース（データ）全体のコピー516を含んでいる。初期化送信ファイル310はさらに、ファイル310の発生前に発生された最新の送信ファイル300のファイル識別子400であるフィールド510と、ファイル310の発生前にローカルデータベース200へ与えられた最新のトランザクションの識別子であるフィールド512を含んでいる。ローカルおよび遠隔データベース200、210中のデータはデータベース200、210に存在するテーブルに割当てられる。データベース200、210は任意の数のテーブルをサポートする。データベースがテーブルを有するとき、初期化送信ファイル310はテーブルに記録された多数の記録を示す各テーブルのフィールドを含んでいる。例えば、ドメイン名データベースはドメインテーブルとネームサーバテーブルを含んでいる。したがって、初期化送信ファイルは、ドメインテーブル中の多数の記録を示すフィールドおよびネームサーバテーブル中の記録の数を示すフィールドを含むことができる。フィールドは例えばテーブル名、テーブルの記録のインデックスに使用されるキー、テーブル中の記録の数を特定する。さらに、初期化送信ファイル310は初期化送信ファイル310のバージョンを示すフィールド、通常は1.0を含んでいる。これらの初期化送信ファイルフィールドは例示として示されたものであり、本発明の実施形態の技術的範囲を限定する目的ではない。任意の有効なフィールドは初期化送信ファイル310に含まれることができる。

【 0 0 3 0 】

初期化送信ファイル310は例えば前述したように、ローカルデータベース200全体の読取り一致コピーを含んでいる。初期化送信ファイル310は t_s と t_f との間の時間 t のポイントでローカルデータベース200と一貫するようになり、ここで t_s は初期化送信ファイル310の発生が開始する時間であり、 t_f は発生が終了する時間である。このようにして、初期化送信ファイル310で行われる唯一の動作は“付加”動作である。即ち初期化送信ファイル310が発生されるとき、時間 t におけるローカルデータベース200全体のコピーは初期化送信ファイル310中に記録される。したがって“付加”動作は初期化送信ファイル310でローカルデータベース200を記録するために行われる。識別子は任意の順番で初期化送信ファイル310に記録されることができる。代わりに、外部の識別子が存在すると、参照されたデータ記録は基準データ記録の前に記録される。

【 0 0 3 1 】

フィールド510、512の追加は、初期化送信ファイル310が発生されながら、初期化転送ファイル310に、発生され遠隔データベース210へ与えられる送信ファイル300の認知を与える。しかしながら、送信ファイル300と初期化送信ファイル310の発生は、発生において他方への依存性がないことに関して相互から分離される。このような構造およびプロセスは送信ファイルの発生と供給が初期化送信ファイル発生が完了されるまで中止される効率

の劣る方法を防止する。本発明の1実施形態のように初期化送信ファイル310を発生しながら、送信ファイル300を発生して供給し続けることにより、送信ファイル300の強力なエラーチェックが行われ、同様に遠隔データベース210における制約、例えば特有の制約または外部の識別子の制約の設定が行われる。制約の設定は遠隔データベース210の関連性のあるモデルを侵害するトランザクションを却下することにより遠隔データベース210のデータの完全性を保護する。例えば特有の制約は同一のキーが一度よりも多数回、データベース210に記憶されることを防止する。

【0032】

図6は本発明の1実施形態にしたがった送信ファイルと初期化送信ファイルの発生の例示的なタイミングチャートである。図6では、送信ファイル300(sf-1乃至s-21)は規則的な時間間隔で発生される。別の実施形態では、送信ファイル300は不規則な時間間隔で発生されてもよい。通常送信ファイルの発生は時間間隔全体を取らない。例えばファイルが5分間の間隔で発生されるならば、ファイルの発生を完了するために5分間全てを取らない。さらに、送信ファイル300が発生されているあいだにローカルデータベース200に変更が生じるならば、これらの変更は次の送信ファイル300で捕捉される。例えば送信ファイルsf-4が12:05:00で発生を開始し、12:05:02で完了するならば、12:05:00と12:05:02の間で生じたローカルデータベース200に対する変更は送信ファイルsf-5(例えば300-5)で捕捉され、これは12:05:00から12:10:00の時間期間を捕捉する。

【0033】

送信ファイル300-5と300-19は図6で示されている。これらのファイルは他のフィールドの中で、ファイル識別子601(sf-5、sf-19)、ファイル発生時間603、トランザクション識別子605(例えば10002)を示している。トランザクション識別子は単調に順序付けされていないことに注意する。前述したように、トランザクション識別子はランダム値を有してもよい。しかしながら、関連されるトランザクション自体はこれらがローカルデータベース200で生じた順序で送信ファイル300に記録される。

【0034】

初期化送信ファイル310の発生と送信ファイル300の発生は分断されてよいので、初期化送信ファイル310は任意の時間に発生されることができる。例えば初期化送信ファイル310は送信ファイル300の発生前、発生中、発生後に発生されてもよい。図6は第4と第5の送信ファイル(例えばsf-4とsf-5)の間の中点で発生される初期化送信ファイル310を示している。

【0035】

1実施形態では、初期化送信ファイル310は他のフィールドの中で、ファイル識別子610(isf-1)、初期化送信ファイルの発生前に発生された最新の送信ファイルのファイル識別子615、初期化送信ファイルの発生前に与えられた最新のトランザクションのトランザクション識別子620を含んでいてもよい。この例では、発生された最新の送信ファイルは送信ファイルsf-4であり、与えられた最新のトランザクションはトランザクション10001である。初期化送信ファイル310は12:07:29で発生611を開始する。送信ファイル300-5(sf-5)の最初の半分のトランザクション、即ちトランザクション10002、10005、10001は、初期化送信ファイル310が発生を開始するとき、ローカルデータベース200へ既に与えられている。したがって、初期化送信ファイル310はこれらのトランザクションを認知し、初期化送信ファイル310でこれらのトランザクションを捕捉する。しかしながら、初期化送信ファイル310は初期化送信ファイルの発生の開始後に生じるその後のトランザクション10003と10004を認知することはできない。

【0036】

初期化送信ファイル310が発生され、送信ファイル300-5で開始する送信ファイルは規則的な間隔で発生され続ける。これらのファイルは遠隔データベース210へ送信され、供給される。

【0037】

初期化送信ファイル310は18番目と19番目の送信ファイル300(sf-18およびsf-19)の発生の中点で、1:15:29で発生を完了し、19番目の送信ファイル300-19の発生に影響しない。

【0038】

遠隔データベース210で初期化送信ファイル310を受信しロードした後、遠隔データベース210は初期化送信ファイル310の発生前に発生された送信ファイルを考慮しない。これは例えば先の送信ファイル300に記録されたローカルデータベース200に対する全ての変化を含んでいる初期化送信ファイル310のためである。この例では、遠隔データベース210は第1乃至第4の送信ファイル(sf-1乃至sf-4)を考慮する必要はない。これらの送信ファイルsf-1乃至sf-4に記録された変更は初期化送信ファイル310にも記録されることができる。これらの先の送信ファイル(sf-1乃至sf-4)は消去され、または代わりにアーカイブされてもよい。同様に、遠隔データベース210は後に発生される送信ファイル300に含まれた初期化送信ファイル310の発生前に与えられたトランザクションを考慮しない。初期化送信ファイル310は初期化送信ファイル310が発生されるときにこれらのトランザクションを含む。例えば遠隔データベース210はこれらのトランザクションのために送信ファイルsf-5の最初の3つのトランザクション10002、10005、10001を考察する必要はない。sf-5に記録されるこれらのトランザクションはまた初期化送信ファイル310に記録されることができる。これらの与えられたトランザクションは消去されるか、代わりにアーカイブされてもよい。

【0039】

図7はローカルデータベースの更新ファイルが発生される本発明の1実施形態のフローチャートである。システムは(705で)ローカルデータベースに対するインクリメント変化に基づいて複数の周期的な更新を発生する。各更新は1以上のトランザクションを含んでいる。システムはその後、(710で)周期的な更新を遠隔データベースへ送信する。定期的な更新が発生される一方で、システムは(715で)開始時間で初期化更新を発生し始める。初期化更新はローカルデータベース全体のバージョンを含んでいる。システムは開始時間前に発生される最新の定期的な更新と、開始時間前に与えられる最新のトランザクションを決定することができる。その後、システムは(725で)初期化更新を遠隔データベースへ送信する。初期化更新は発生される最新の定期的な更新に関連される更新識別子と、与えられる最新のトランザクションに関連するトランザクション識別子とを含んでいる。

【0040】

例えば、OLTP140は(705で)幾つかの規則的または非規則的な時間間隔で送信ファイル300を発生する。OLTP140はその後(710で)、遠隔データベース210へ送信ファイル300を送信する。送信ファイル300が発生されるあいだに、OLTP140は(715で)開始時間611で初期化送信ファイル310を発生し始める。初期化送信ファイル310はローカルデータベース200のコピーを含んでいる。OLTP140はその後、(720で)初期化送信ファイル310の発生のため開始時間611前に発生される最新の送信ファイル300と、初期化送信ファイル310の発生のため開始時間611前に与えられる最新のトランザクションを決定する。OLTP140はその後、(725で)初期化送信ファイル310を遠隔データベース210へ送信する。初期化送信ファイル310は発生された最新の送信ファイル300に関連する送信ファイル識別子615と、与えられた最新のトランザクションに関連するトランザクション識別子620を含んでいる。

【0041】

図8は遠隔データベースがローカルデータベースから更新ファイルを受信する本発明の1実施形態のフローチャートである。システムは(805で)複数の定期的な更新を受信する。各更新は1以上のトランザクションを含んでいる。定期的な更新は個別にまたはバッチで受信される。システムは(810で)ある時間に、初期化更新を受信する。初期化更新はローカルデータベース全体のバージョンを含んでいる。システムは(815で)最新の定期的な更新識別子と、最新のトランザクション識別子を初期化更新から読取る。システム

はその後、(820で)更新識別子に関連する最新の定期的な更新と、トランザクション識別子に関連する最新のトランザクションを決定する。定期的な更新およびトランザクションは初期化更新の発生前にそれぞれ発生され与えられた最新のものである。システムは(825で)対応する定期的な更新の残りの与えられていないトランザクションを遠隔データベースに適用する。システムはその後、(830で)最新の定期的な更新後に発生された残りの定期的な更新を遠隔データベースへ供給する。初期化更新を有効に供給することは任意の先に失われた定期的な更新の埋め合わせをする。

【0042】

例えば、LUE100は(805で)幾つかの定期的または非定期的な時間間隔で送信ファイル300を受信する。送信ファイル300は個別にまたはバッチで受信される。LUE100は(810で)ある時間に初期化送信ファイル310を受信する。LUE100は(815で)送信ファイル識別子615と、トランザクション識別子620を初期化送信ファイル310から読取る。その後、LUE100は(820で)送信ファイル識別子615に関連した送信ファイル300とトランザクション識別子620と関連するトランザクション605を決定する。送信ファイルとトランザクションは初期化送信ファイル310の発生前にそれぞれ発生され与えられた最新のものである。LUE100は、(825で)対応する送信ファイル300の残りの与えられていないトランザクション605を遠隔データベース210へ適用する。LUE100はその後、(830で)最新の送信ファイルsf-4後に、残りの送信ファイル300を遠隔データベース210へ適用する。

【0043】

別の実施形態では、例えばLUE100は遠隔データベース210へ与えられないおよび/または初期化送信ファイル発生時間611前の発生時間603を有する送信ファイル300を廃棄またはアーカイブする。廃棄またはアーカイブされる送信ファイル300は送信ファイル識別子615に関連される送信ファイルsf-4を含んでいる。

【0044】

初期化送信ファイル310が与えられた後、既に遠隔データベース210へ与えられている任意の後の送信ファイル300は、遠隔データベース210が初期化送信ファイル310との読取り一致になるために失われることが理解されよう。したがって、これらの後の送信ファイル300は再度与えられる。

【0045】

本発明の1実施形態では、送信ファイル300と初期化送信ファイル310は、承認なしに、即ちファイルが適切に受信されたことを示すためのACK/NACK信号なしにローカルデータベース200から遠隔データベース210へ送信されることができる。これは有効にACK/NACK信号が生成するオーバーヘッドを減少する。

【0046】

別の実施形態では、ACK/NACK信号はファイルの適切な受信を示すために遠隔データベース210から送信される。この実施形態では、ACK/NACK信号は確實ではない通信によるシステムで送信される。

【0047】

図9は、ローカルデータベースから送信され、遠隔データベースで受信される更新ファイルをシステムが確認する本発明の別の実施形態のフローチャートである。ここで、システムは(905で)複数の定期的な更新を送信する。各更新は1以上のトランザクションを含んでいてもよい。定期的な更新は個別またはバッチで送信される。システムは(910で)ある時間に初期化更新を送信し、初期化更新を遠隔データベースへ与える。初期化更新はローカルデータベース全体のバージョンを含んでいる。システムは最初に(915で)データベースを比較することによりローカルおよび遠隔データベース間の不一致を識別する。システムは(920で)不一致が有効であるかエラーを有するかを決定する。システムはその後(925で)本発明の実施形態にしたがって定期的な更新を遠隔データベースへ与える。この実施形態はローカルデータベースからの更新の受信結果として遠隔データベースにエラーが存在しないことを確実にする。

【0048】

10

20

30

40

50

例えば、O L T P 140は(905で)幾つかの規則的または不規則の時間間隔で送信ファイル300を遠隔データベース210へ送信する。送信ファイル300は個別またはバッチで送信される。O L T P 140は(910で)ある時間に初期化送信ファイル310をL U E 100へ送信し、L U E 100は初期化送信ファイル310を遠隔データベース210へ送信する。O L T P 140はローカルデータベース200を遠隔データベース210と比較し、(915で)それらの間の不一致を識別する。O L T P 140はその後(920で)不一致が有効またはエラーがあるかを決定する。O L T P 140はその後本発明の1実施形態にしたがって、L U E 100に(925で)送信ファイル300を遠隔データベース210へ与えることを通知する。L U E 100はその後、送信ファイル300を遠隔データベース210へ与える。

【0049】

10

別の実施形態では、システムは不一致を識別し、確認する前に送信ファイルと初期化送信ファイルの両者を供給する。その代わりに、システムは不一致を識別し、確認した後に送信ファイルと初期化送信ファイルの両者を供給することができる。

【0050】

確認プロセスは送信されたデータを目的地へ与える目的で、ソースから目的地へネットワークで送信された任意のデータで行われることが理解されるであろう。

【0051】

図10Aは、本発明にしたがった送信ファイルと初期化送信ファイルの確認の1実施形態のフローチャートである。複数の定期的な更新と初期化更新を遠隔データベースへ送信した後、システムはこれらの更新を確認する。各更新はローカルデータベースで行われた1以上のトランザクションを含んでいる。各トランザクションは1以上の事象を含んでいる。事象はデータベース中のデータに関し、データベースのアクションまたは発生、例えば付加、変更、消去などである。

20

【0052】

最初に、システムは(1000で)遠隔データベース中の記録を、ローカルデータベース中の対応する記録と比較する。システムは(1005で)遠隔およびローカルデータベース間の不一致を説明する例外を発生し、例外は各不一致に対して発生される。不一致は同一の記録の2つのバージョン間の少なくとも1つのデータ値の任意の差である。例えば、ローカルデータベース中のデータ記録は(12345, x y z . c o m , 123 . 234 . 345)である。同一であると想定される遠隔データベース中の対応するデータ記録は(12345, a b c . c o m , 123 . 234 . 345)である。したがって、記録の第2のデータ値には不一致が存在する。したがって、本発明の1実施形態はこの不一致を説明する例外を発生する。例外は不一致が存在することを単に示し、不一致の位置を特定し、不一致の2つのデータ値間の差を説明すること等によって不一致を記述する。2つの記録が同一のデータを含んでいることが想定されるならば、ローカルデータベースのデータ記録は遠隔データベースのデータ記録に対応する(その逆もある)。

30

【0053】

不一致は記録または、その全体の記録の1以上のデータ値の差を指していることが理解されよう。

【0054】

40

システムは(1010で)例外識別子を各例外に関連付け、例外識別子は記録の識別子と関連される。例えばデータ記録(12345, x y z . c o m , 123 . 234 . 345)は、識別子d10を有する。したがって例外識別子もまたd10である。各例外は多数の例外(または不一致)タイプのうちの任意の1つに属すとして分類される。例外リストは例外タイプと、そこで区分された例外の例外識別子を含むように形成される。例外リストおよび異なる例外タイプについては後に詳細に説明する。システムはまた(1015で)事象識別子を更新の各事象に関連付け、事象識別子は記録の識別子と関連付けされる。例えばデータ記録(12345, x y z . c o m , 123 . 234 . 345)は識別子d10を有する。したがって、事象識別子もまたd10である。更新の各事象は事象経歴から発見される。事象経歴は時間期間にわたってローカルデータベース中の記録について行われる

50

事象のリスト等である。事象経歴を後に詳細に説明される。

【 0 0 5 5 】

システムはその後、(1020で)記録の更新が有効であるか否かを決定する。図10Bは確認決定の1実施形態のフローチャートである。この決定は以下に行われる。各事象は(1022で)各例外と比較される。各例外が(1024で)事象により正当化されないならば、更新は(1026で)有効として指定され、更新は遠隔データベースに供給される。そうでなければ、各例外が(1024で)事象により正当化されないならば、更新は(1028で)無効として指定され、例外はエラーとしてログされる。例外は事象識別子が例外識別子に対応し、関連する事象が例外タイプに関連される事象の有効なシーケンスに対応するとき正当化される。有効なシーケンスについては後に詳細に説明する。例外が正当化されるならば、システムは例外識別子を例外リストから除去する。正当化された例外は不一致が有効なものであることを示し、例えば遠隔データベースは更新をまだ受信していないが、更新が受信されるときにローカルデータベースと一致する。

10

【 0 0 5 6 】

確認中、システムは定期的および初期化更新の潜在的エラーまたは故障を識別する。システムはこれらの更新が構造的および意味的に正しく、これらの更新が例外を発生せずまたはそうでなければ不適切に停止せずに適用され、ローカルおよび遠隔データベース間の比較が正確にエラーを検出し、高いプロフィールデータが偶然に消去されないことを確実にする。システムは定期的および初期化更新が適切に遠隔データベースに供給されることを確実にする。

20

【 0 0 5 7 】

多数のエラーは有効に、確認中に更新を遠隔データベースへ供給することを試みるにより発見される。例えばデータ中心エラー、オブジェクトが既に遠隔データベースに存在するという警告、または外部の識別子の侵害が存在するという警告は、供給を試みる期間中に発見される。したがって、本発明の実施形態の確認プロセスの実行後、システムはこれらの更新中を遠隔データベースに適用しようとする。この試みは失敗し、更新を無効にする付加的なエラーが更新中に存在することを示す。したがって、これらの更新を遠隔データベースに適用する試みはさらに行われない。

【 0 0 5 8 】

別の実施形態では、確認を実行する前に、少なくとも1つの更新を供給する試みが行われる。試みが失敗したならば、確認はスキップされ、更新は廃棄される。他方で、試みが成功したならば、確認が実行され、有効な更新が維持され、無効な更新は不一致のためにログされる。

30

【 0 0 5 9 】

例示的な実施形態では、OLTP140は送信ファイル300と初期化送信ファイル310が適切に遠隔データベース210へ供給されることを確実にするために、送信ファイル300と初期化送信ファイル310を確認する。

【 0 0 6 0 】

別の実施形態では、ネットワークコンピュータ121、LUE100、または既存のシステムの任意の組み合わせが確認を実行する。

40

【 0 0 6 1 】

図10Aを参照すると、OLTP140はローカルデータベース200と遠隔データベース210とを比較し、それらの間の任意の例外(または不一致)を決定する。例外は3つのタイプを含んでおり、即ちそれらはデータがローカルデータベース200ではなく遠隔データベース210に存在し、データが遠隔データベース210ではなくローカルデータベース200に存在し、または対応するデータがローカルデータベース200と遠隔データベース210に存在するがデータが異なるタイプである。勿論、対応するデータはローカルデータベース200と遠隔データベース210とに存在し、データは同一であり、この場合データは有効であると考えられ、したがってさらにOLTP140による処理を必要としない。

【 0 0 6 2 】

50

不一致は記録またはその全体の記録の1以上のデータ値を指していることが理解されよう。

【0063】

したがって、OLTP140は(1000で)ローカルデータベース200と遠隔データベース210の対応する記録を比較する。OLTP140は(1005で)遠隔データベース210の記録と、ローカルデータベース200の記録との不一致を記述する例外を発生し、ここで例外は各不一致に対して発生される。OLTP140は(1010で)例外識別子を各例外に関連付け、各例外識別子は記録の識別子に関連される。例外リストは例外タイプと、例外タイプに属す例外の例外識別子とを含むように形成される。1実施形態では、例外はそれが第1の例外タイプに属すならば、“リスト1”例外(または不一致)として指定され、例外が第2の例外タイプに属すならば、“リスト2”例外、または例外が第3の例外タイプに属すならば、“リスト3”例外として指定される。図11は例示的な例外リスト1140を示している。

10

【0064】

例えばローカルデータベース20に対する変更と遠隔データベース210に供給される更新との間の時間遅延のために全ての3つのタイプの例外が正当に生じるので、例外リストの例外識別子の存在は送信ファイル300または初期化送信ファイル310が悪いことを示唆しないことが理解されよう。このような遅延は例えばネットワークの混雑により生じる可能性がある。このようにして、確認はエラーのあるデータから正当性を取り除く機構を与える。

【0065】

20

初期化送信ファイル310に対しては、OLTP140はデータベース200と210の両方で双方向フルテーブル走査を実行することによりローカルデータベース200と遠隔データベース210とを比較する。即ち、ローカルデータベース200中の全てのデータは遠隔データベース210中の全てのデータに対して比較される。その後、遠隔データベース210中の全てのデータはローカルデータベース200中の全てのデータに対して比較される。これは全ての不一致を発見するためとデータベース200、210の徹底的な比較を与える利点がある。

【0066】

送信ファイル300では、OLTP140は送信ファイル300に記録されているローカルデータベース200と遠隔データベース210のデータ記録だけを比較する。これはターゲットとする不一致を発見するための迅速な問い合わせを与える利点がある。

30

【0067】

その代わりに、初期化送信ファイル310および/または送信ファイル300のデータのランダムなサンプリングが行われる。OLTP140はその後、ローカルデータベース200と遠隔データベース210のサンプルされたデータをランダムに比較する。

【0068】

例外リスト1140は、遠隔データベース210と一貫しないローカルデータベース200に対する失った事象、例えば付加(add)、変形(mod)、消去(del)に対応する。これらの候補事象を識別するために、OLTP140はローカルデータベース200に与えられた最近のトランザクションを検査する。通常、与えられる各トランザクションに対して、エントリはローカルデータベース200に記憶されたログテーブル中に行われる。エントリは変更された記録に対する識別子と、記録を変更したトランザクション(または事象)(例えばadd、modおよび/またはdel事象)、トランザクションの順序を示すログシーケンス番号等を含んでいてもよい。

40

【0069】

例示的なログテーブル1100が図11で示されている。この例では、送信ファイル300はログテーブル1100で示されているトランザクション1108-1114を含んでいる。第1のエントリ1101は第1のトランザクション1108で、データn1、n2(ネームサーバ)が識別子d1に関連されるデータ(ドメイン)に付加されたことを示している。識別子はd1であり、事象は“add”であり、ログシーケンス番号は11526である。同様に、第2のエントリ1102は第2のトランザクション1109で、データn8とn9が識別子d2と関連さ

50

れるデータに付加されたことを示している。第3のエントリ1103は第3のトランザクション1110で、識別子d3と関連されるデータが消去されたことを示している。第4のエントリ1104は第4のトランザクション1111で、識別子d1と関連されるデータがデータn5を付加するように変更されたことを示している。第5のトランザクション1112では、第5のエントリ1105は、データn6とn7が識別子d3と関連されるデータに付加されたことを示している。第6のトランザクション1113では、第6のエントリ1106は、識別子d4に関連されるデータがデータn3を除去するために変更されたことを示している。第R番目のエントリ1107は、第R番目のトランザクション1114で、識別子d5に関連されるデータが消去されたことを示している。

【0070】

したがって、図10Aに示されているように、OLTP140は(1015で)事象識別子を更新の各事象に関連付け、事象識別子は記録の事象に関連されている。更新の各事象は事象経歴から発見される。事象識別子によりインデックスされ順序付けされた事象経歴はログテーブル1100から発生されることができ、例示的な事象経歴1120は図11で示されている。ここで、ログテーブル1100の第1および第4のエントリ1101、1104は識別子d1に関連されるデータに対する変更を示している。したがって事象経歴1120はd1識別子1121と、識別子d1に関連されるデータにおいて“add”とその後に“mod”が行われる2つの事象1126を含んでいる。第2のエントリ1102は識別子d2に関連されるデータに対する変更を示している。したがって事象経歴1120はd2識別子1122と“add”事象1127を含んでいる。事象経歴1120はd3識別子1123と、“del”とその後に“mod”が行われる2つの事象1128を含み、第3および第5のエントリ1103と1105を示し、識別子d3に関連するデータへの変更を含んでいる。第6のエントリ1106は識別子d4に関連されるデータに対する変更を示している。したがって、事象経歴1120はd4識別子1124と“mod”事象1129を含んでいる。第Rのエントリ1107は識別子d5に関連されるデータに対する変更を示し、事象経歴1120はd5識別子1125と“del”事象1130を含んでいる。識別子1121-1125はd1乃至d5に順序付けされる。

【0071】

図10Aを再度参照すると、OLTP140は(1020で)更新が有効であるか否かを決定する。この決定は例えば図10Bの実施形態にしたがって行われる。最初に、OLTP140は(1022で)どの識別子に対応するかを決定するために事象識別子1121-1125を例外識別子1140と比較する。例えば図11では、事象経歴1120のd1事象識別子1121は例外リスト1140の“リスト2”のd1例外識別子に対応する。対応する事象および例外を発見後、OLTP140は(1024で)事象が例外を正当化するか否かを決定する。正当化は以下のように行われる。事象経歴1120の各事象識別子1121-1125では、OLTP140は事象経歴1120の事象1126-1130の各システムが有効であるか否かを決定する。これは例えば各例外識別子が属する例外タイプを決定するために例外リスト1140を検査し、その例外タイプの事象の有効なシーケンスが何であるべきかを決定し、その後対応する事象識別子と事象の事象識別子のシーケンスの事象経歴1120をサーチすることによって行われる。各例外タイプの有効なシーケンスについて以下詳細にする。事象経歴1120の事象1126-1130のシーケンスが有効なシーケンスと一致するならば、対応する事象識別子1121-1125は有効なシーケンスを有する。このように、例外識別子に関連する例外は正当化される。その事象を含んでいる対応するトランザクション1108-1114は正当化されたものでありエラーがない。この場合、OLTP140は例外リスト1140から例外識別子を除去する。

【0072】

“リスト1”例外タイプの事象の有効なシーケンスは(mod)*(del)である。このシーケンスはゼロ以上の“mod”事象と、それに後続する“del”事象と、その後何かが続くシーケンスを含んでいる。“リスト1”例外タイプは遠隔データベース210に存在するデータに対応するが、ローカルデータベース200に存在するデータには対応しない。この場合、データはローカルデータベース200から最近消去され、トランザクションはまだ送信ファイル300に書き込まれていない。したがって送信ファイル300はまだ遠隔

10

20

30

40

50

データベース210に供給されていない。それ故データは遠隔データベース210に存在しない。これは幾つかのポイントで、送信ファイル300が発生されて遠隔データベース210に適用されることが予測されるので、正当化の不一致であると考えられる。任意のこのようなシーケンス1126-1130が例外リスト1140のリスト1の例外識別子の事象経歴1120で発見されるならば、対応するトランザクションは有効であると考えられてもよい。

【0073】

例えば、図11では、d5識別子1125とその関連されるデータは、ログテーブル1100の第Rのエントリ1114で示されているようにローカルデータベース200から消去され、事象経歴1120でインデックスされる。確認のときに、d5はローカルデータベース200から消去されるが、遠隔データベース210から消去されない。例外リスト1140はリスト1に識別子d5を含んでいる。事象経歴1120にしたがって、d5識別子1125に関連される事象1130は“del”である。OLTP140は“リスト1”例外タイプの有効なシーケンス、即ち(mod)* (del)を事象経歴1120のd5事象1130に対して比較する。“リスト1”の有効なシーケンスと事象1130が一致するので、識別子d5に関連する消去トランザクション1114は正当であり、エラーとは考えられない。したがって識別子d5は例外リスト1140から除去されることができる。

【0074】

“リスト2”例外タイプの事象の有効なシーケンスは(add)である。このシーケンスは“ add ”事象と、その後になんかが続くシーケンスを含んでいる。“リスト2”例外タイプはローカルデータベース200に存在するデータに対応するが、遠隔データベース210に存在するデータには対応しない。この場合、データはローカルデータベース200へ最近付加され、トランザクションはまだ送信ファイル300に書き込まれていない。送信ファイル300はまだ遠隔データベース210に適用されていない。データは遠隔データベース210に存在しない。これは幾つかのポイントで、送信ファイル300が発生されて遠隔データベース210に適用されることが予測されるので、正当化の不一致であると考えられる。したがって、任意のこのようなシーケンス1126-1130が例外リスト1140のリスト2の例外識別子の事象経歴1120で発見されるならば、対応するトランザクションは有効であると考えられる。

【0075】

図11を参照すると、d1、d2識別子1121、1123は例えばローカルデータベース200に最初に付加されたデータに関連されている。事象1126、1127のシーケンスは“ add ”事象で開始するので、d1、d2識別子1121、1123は“リスト2”例外タイプの有効なシーケンスと一致する。したがって、これらの識別子を含んでいるトランザクション1108、1109は有効であると考えられ、識別子d1、d2は例外リスト1140から除去される。d3識別子1123もそのシーケンス1128で“ add ”事象を含むことに注意する必要がある。しかしながら、“ add ”事象はシーケンス1128で最初ではない。したがってシーケンス1128は“リスト2”タイプとして適格とされない。さらにd3は例外リスト1140のリスト2で指定されないので、OLTP140はそれをリスト2の有効なシーケンスでチェックしない。

【0076】

“リスト3”例外タイプの事象の有効なシーケンスは(del)*(add)または(mod)である。これらのシーケンスは“ mod ”事象と、それに後続し、その後になんかが続く“ add ”事象またはその後になんかが続く“ mod ”事象を含んでいる。“リスト3”例外タイプはデータベース200、210の両者に存在するデータに対応するが、異なっている。この場合、データはローカルデータベース200で最近変更され、トランザクションはまだ送信ファイル300に書き込まれていない。送信ファイル300はまだ遠隔データベース210に適用されていない。識別子に関連されているデータは遠隔データベース210でまだ変更されていない。これは幾つかのポイントで、送信ファイル300が発生されて遠隔データベース210に適用されることが予測されるので、正当化の不一致であると考えられる。したがって任意のこのようなシーケンス1126-1130が例外リスト1140のリスト3の例外識別

子の事象経歴1120で発見されるならば、対応するトランザクションは有効であると考えられる。

【 0 0 7 7 】

例えば、図 1 1 では、d 3、d 4 識別子1123、1124はローカルデータベース200で変更されたデータに関連されている。d 3 識別子1123の場合、d 3 識別子1123とそのデータは最初に消去され、その後新しいデータを付加され、それによって事象1128のシーケンスは“ d e l ”とそれに後続する“ a d d ”を含んでいる。d 4 識別子1124の場合、d 4 データはデータを除去するように変更され、それによって事象1129のシーケンスは“ m o d ”を含んでいる。事象1128、1129のこれらのシーケンスは“ リスト 3 ”例外タイプの有効なシーケンスと一致するので、これらの対応するトランザクション1110、1112、1113は有効であると考慮され、識別子 d 3、d 4 は例外リスト1140から除去される。

10

【 0 0 7 8 】

図 1 0 B を参照すると、例外リスト1140の識別子により示されている全ての例外が（1024で）事象により正当化されるならば、即ち例外リスト1140がエンプティならば、O L T P 140は（1026で）送信ファイル300または初期化送信ファイル310を有効として指定し、L U E 100に送信ファイル300または初期化送信ファイル310を遠隔データベース210へ適用するように通知する。L U E 100はその後、送信ファイル300または初期化送信ファイル310を遠隔データベース210へ供給する。

【 0 0 7 9 】

反対に、全ての例外が（1024で）事象により正当化されないならば、即ち例外リスト1140がエンプティではないならば、残りの例外は（1028で）送信ファイル300または初期化送信ファイル310を無効として指定し、エラーをエラーファイルにログする。

20

【 0 0 8 0 】

別の実施形態では、例えば送信ファイル300または初期化送信ファイル310は無効として指定されるならば、予め定められた期間後、O L T P 140は不一致が本当にエラーであることを確実にするため無効な送信ファイル300または初期化送信ファイル310で確認プロセスを反復する。この予め定められた遅延は任意の低速度の送信ファイル300、310とデータベース200、210を送信するためのネットワークのより多くの時間と、読取りを一貫するためのより多くの時間を可能にする。

【 0 0 8 1 】

30

本発明の 1 実施形態では、遠隔データベース210のデータは大きな時間期間によりローカルデータベース200のデータを“ 遅延 ”する。したがって、データベース200、210と検出エラーを比較するため、データベース200、210は時間の同一点で読取りを一貫され、したがってこれらは相互の正確なコピーである。通常、遠隔データベース210はローカルデータベース200へロールフォワードされ、ここで遠隔データベース210のデータは基本的にローカルデータベース200のデータと同じにされる。

【 0 0 8 2 】

したがって、変更の速度を上げるため、任意の現在発生された初期化送信ファイル310およびその後の送信ファイル300は確認の開始前に遠隔データベース210へ供給される。このようにして、不一致の数は非常に減少される。送信ファイル300と310のこのバッチ処理はチャンキングと呼ばれる。チャンクのこれらの送信ファイル300と310の最初および最後はそれぞれ低および高い透かしマークと呼ばれる。最初のチャンクと呼ばれる第 1 のチャンクは初期化送信ファイル310を含んでいる。終端チャンクと呼ばれる全ての次のチャンクは送信ファイル300だけを含んでいる。

40

【 0 0 8 3 】

チャンキングは分離した確認ではなくグループでの確認を行う。したがって、エラーがチャンクで検出されるならば、チャンク全体はエラーが生じた単なる送信ファイル300または初期化送信ファイル310ではなく無効として指定される。

【 0 0 8 4 】

本発明の実施形態の機構および方法は、実施形態の教示にしたがってプログラムされた

50

汎用目的のマイクロプロセッサを使用して実行されよう。本発明の実施形態はしたがって、マシンの読取り可能な媒体を含んでおり、これは本発明の実施形態による方法を実行するためプロセッサのプログラムに使用される命令を含んでいる。この媒体はフロッピー（Ｒ）ディスク、光ディスク、ＣＤ－ＲＯＭを含む任意のタイプのディスクを含んでいるが、これらに限定されない。

【 ０ ０ ８ ５ 】

本発明の幾つかの実施形態をここで特別に示し説明した。しかしながら、本発明の変更および変形は本発明の技術的範囲を逸脱せずに、前述の考察により、特許請求の範囲でカバーされることが認識されるであろう。

【図面の簡単な説明】

10

【 ０ ０ ８ ６ 】

【図 １】本発明の 1 実施形態にしたがったシステムのブロック図。

【図 ２】本発明の 1 実施形態にしたがったシステムハブのブロック図。

【図 ３】本発明の 1 実施形態にしたがったローカルデータベースから遠隔データベースへのデータベース更新の例示的な送信を示す図。

【図 ４】本発明の 1 実施形態にしたがった送信ファイルを示す図。

【図 ５】本発明の 1 実施形態にしたがった初期化送信ファイルを示す図。

【図 ６】本発明の 1 実施形態にしたがった送信ファイルと初期化送信ファイルの発生のタイミングチャート。

【図 ７】ローカルデータベースの更新ファイルが発生される本発明の 1 実施形態のフローチャート。

20

【図 ８】遠隔データベースがローカルデータベースから更新ファイルを受信する本発明の別の実施形態のフローチャート。

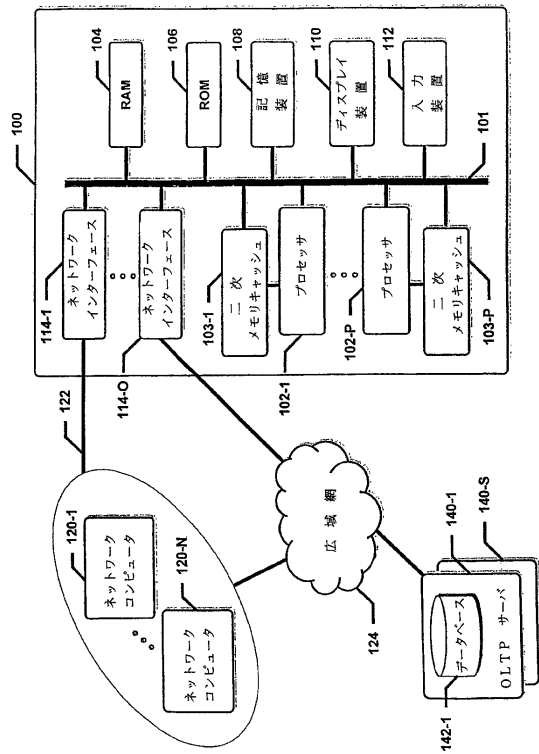
【図 ９】遠隔データベースがローカルデータベースから更新ファイルを受信し、確認する本発明の別の実施形態のフローチャート。

【図 １ ０ Ａ】更新ファイルが確認される本発明の 1 実施形態のフローチャート。

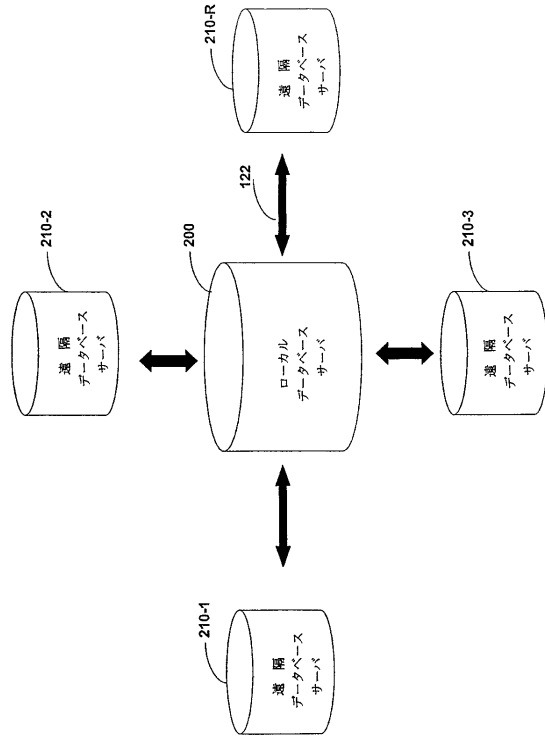
【図 １ ０ Ｂ】更新ファイルが確認される本発明の別の実施形態のフローチャート。

【図 １ １】本発明の 1 実施形態にしたがった更新ファイル確認を示す図。

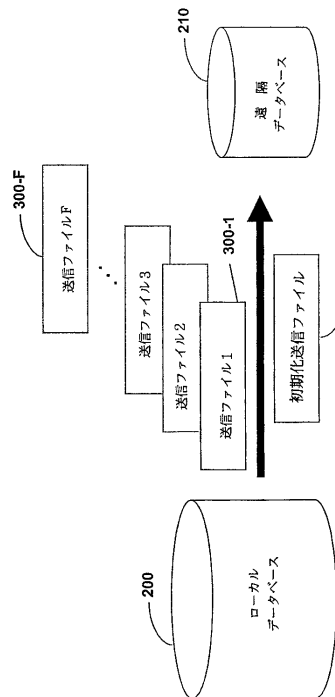
【図 1】



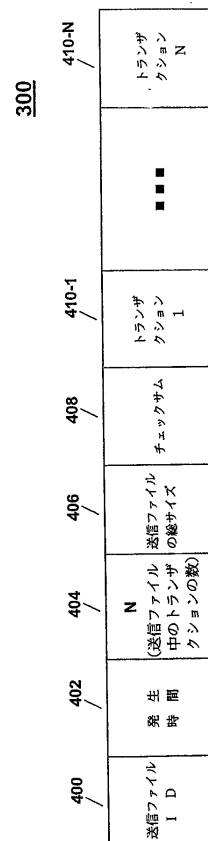
【図 2】



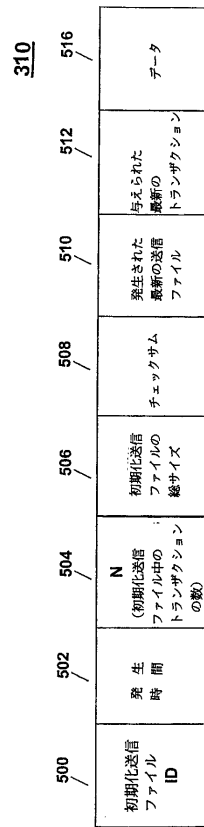
【図 3】



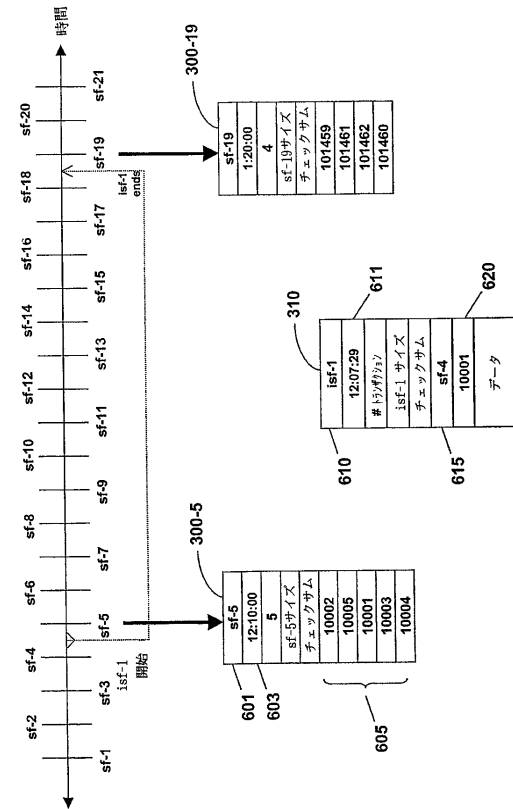
【図 4】



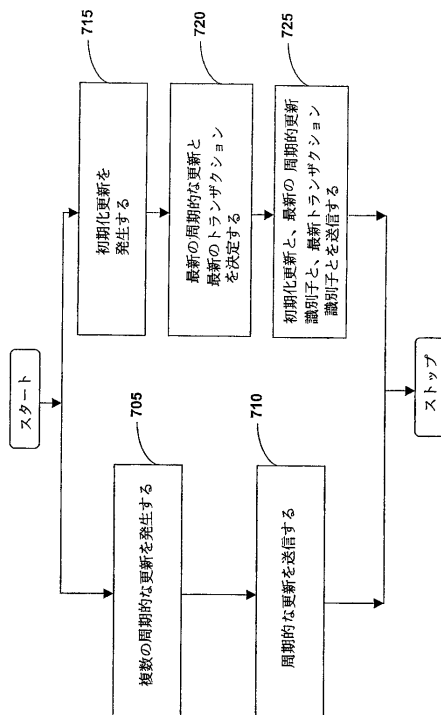
【図 5】



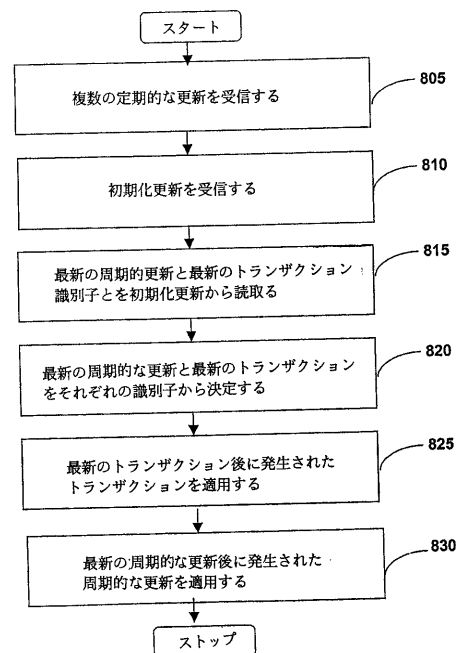
【図 6】



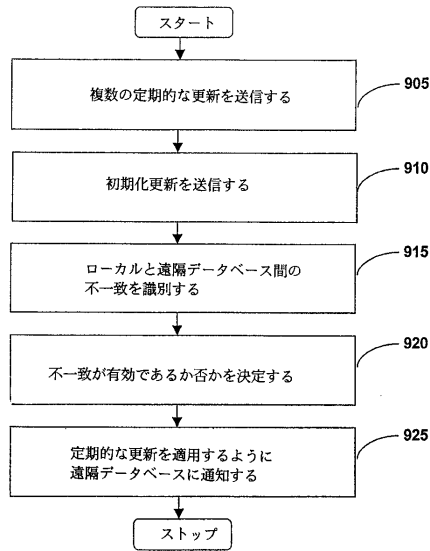
【図 7】



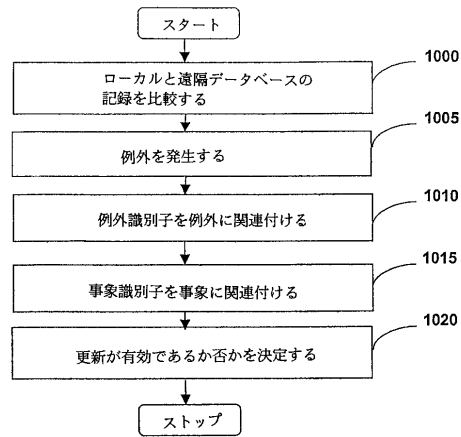
【図 8】



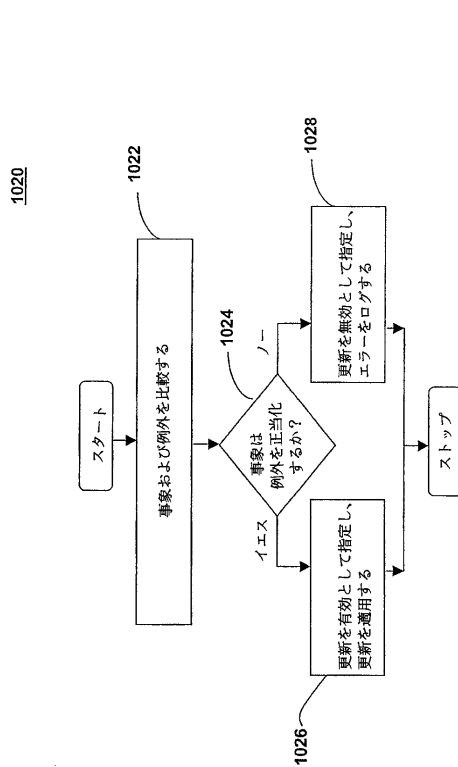
【図 9】



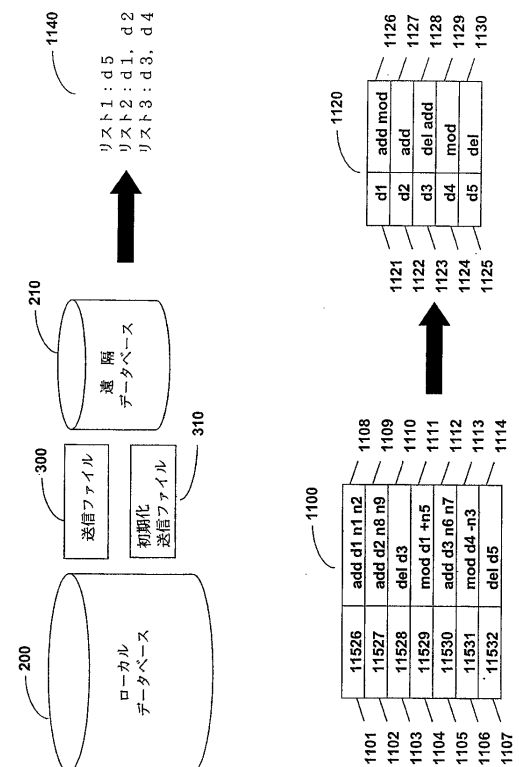
【図 10 A】



【図 10 B】



【図 11】



フロントページの続き

- (72)発明者 バロー、アリストトル・ニコラス
アメリカ合衆国、バージニア州 22124、オークトン、マーシャル・レイク・ドライブ 27
79
- (72)発明者 ハワース、ウィリアム・フレデリック・ジュニア
アメリカ合衆国、メリーランド州 21405、シャーウッド・フォレスト、ノッティンガム・ロ
ード 246
- (72)発明者 マクミレン、ブラドレイ・トーマス
アメリカ合衆国、バージニア州 22101、マクリーン、メイフラワー・ドライブ 1343

審査官 工藤 嘉晃

- (56)参考文献 特開2001-290689(JP, A)
特開2001-236257(JP, A)
特開2000-057030(JP, A)
特開平11-327985(JP, A)

(58)調査した分野(Int.Cl., DB名)

G06F 12/00

G06F 15/00

JSTPlus(JDreamII)